

Calibration and uncertainty quantification using multivariate simulator output

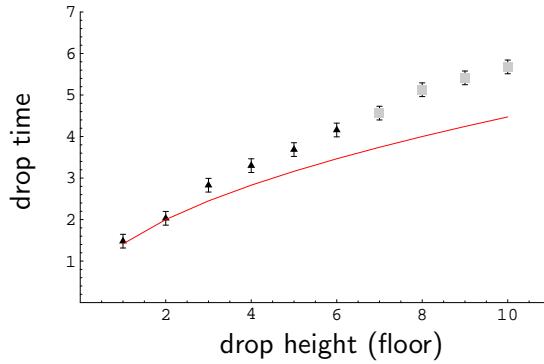
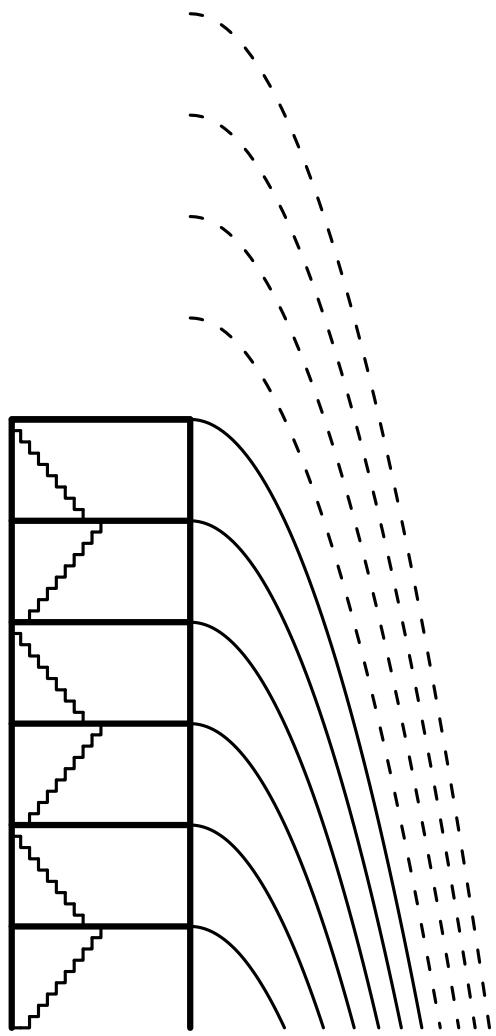
LA-UR 04-7145

Dave Higdon, Statistical Sciences Group, LANL

Jim Gattiker, Statistical Sciences Group, LANL

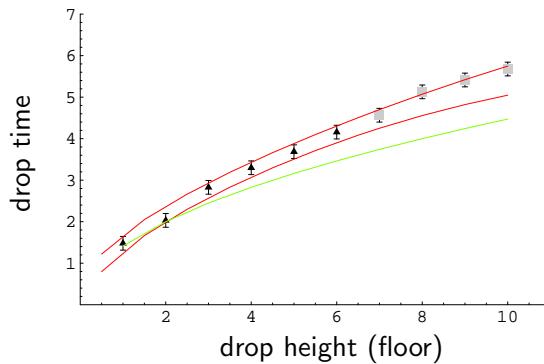
Brian Williams, Statistical Sciences Group, LANL

Inference combining a physics model with experimental data

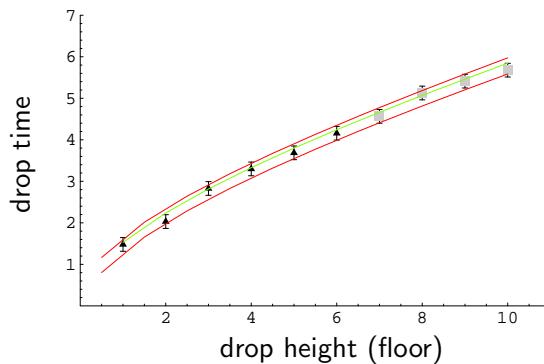


Data generated from model:
$$\frac{d^2z}{dt^2} = -1 - .3\frac{dz}{dt} + \epsilon$$

simulation model:
$$\frac{d^2z}{dt^2} = -1$$



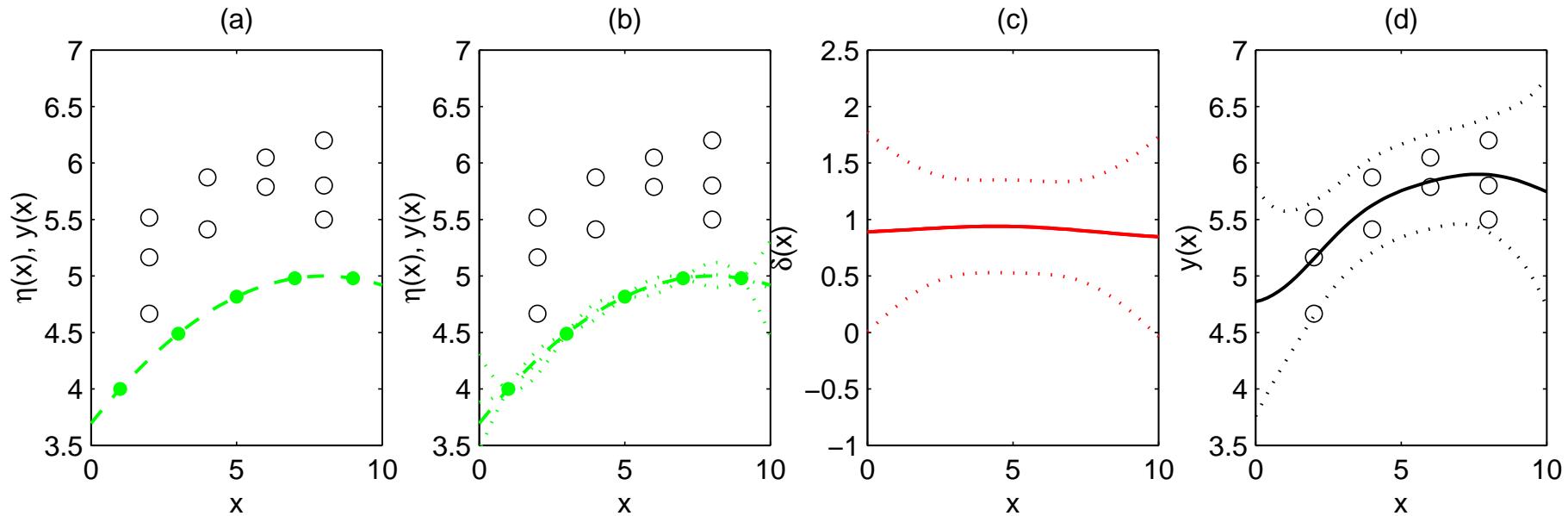
statistical model:
$$y(z) = \eta(z) + \delta(z) + \epsilon$$



Improved physics model:
$$\frac{d^2z}{dt^2} = -1 - \theta\frac{dz}{dt} + \epsilon$$

statistical model:
$$y(z) = \eta(z, \theta) + \delta(z) + \epsilon$$

Basic formulation – borrows from Kennedy and O'Hagan (2001)

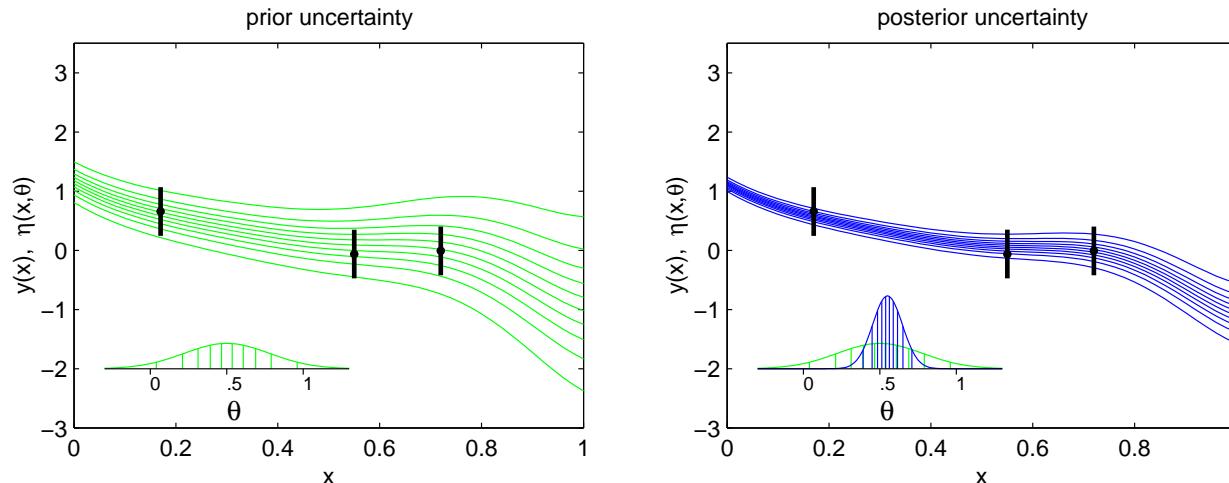


- x experimental conditions
- θ calibration parameters
- $\zeta(x)$ true physical system response given inputs x
- $\eta(x, \theta)$ simulator response at x and θ .
- $y(x)$ experimental observation of the physical system
- $\delta(x)$ discrepancy between $\zeta(x)$ and $\eta(x, \theta)$
may be decomposed into numerical error and bias
- $e(x)$ observation error of the experimental data

$$y(x) = \zeta(x) + e(x)$$

$$y(x) = \eta(x, \theta) + \delta(x) + e(x)$$

A Bayesian approach for combining simulations and experimental data for forecasting, calibration and uncertainty quantification



- A simple example...

x	model or system inputs
θ	model calibration parameters
$\zeta(x)$	true physical system response given inputs x
$\eta(x, \theta)$	simulator response at x and θ .
$y(x)$	experimental observation of the physical system
$e(x)$	observation error of the experimental data

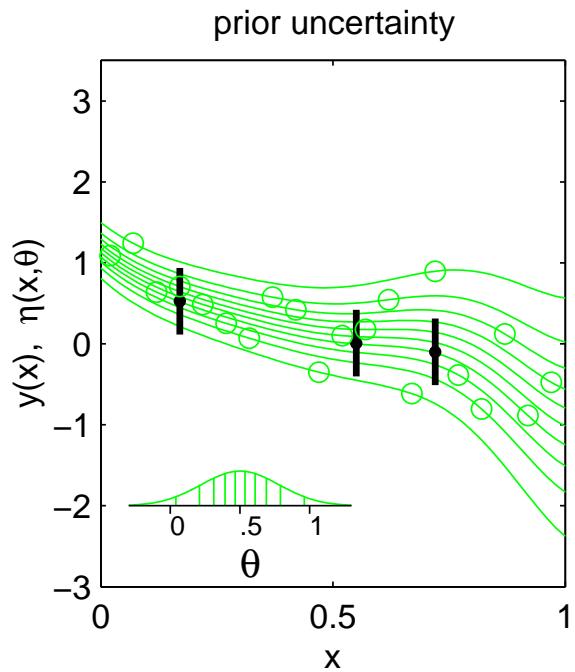
Assume:

$$\begin{aligned} y(x) &= \zeta(x) + e(x) \\ &= \eta(x, \theta) + e(x) \quad \theta \text{ unknown.} \end{aligned}$$

Standard Bayesian estimation gives: $\pi(\theta|y(x)) \propto L(y(x)|\eta(x, \theta)) \times \pi(\theta)$

Accounting for limited simulator runs

- Borrows from Kennedy and O'Hagan (2001).

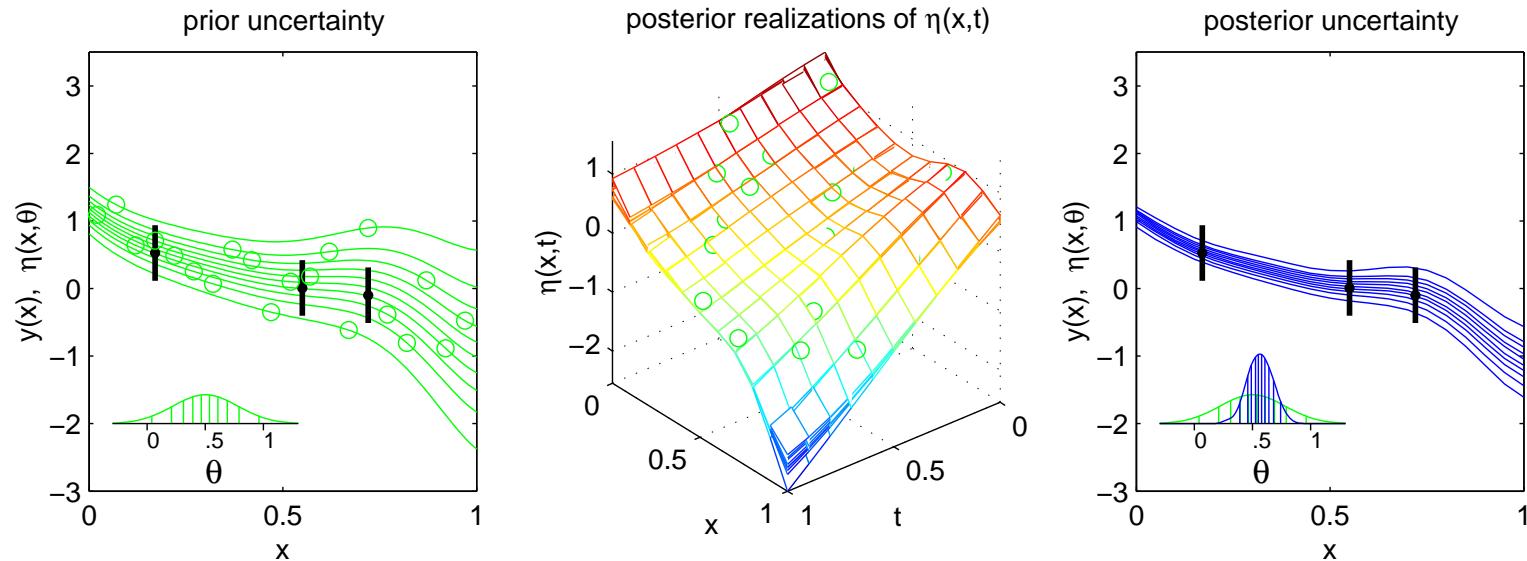


- x model or system inputs
 θ calibration parameters
 $\zeta(x)$ true physical system response given inputs x
 $\eta(x, \theta)$ simulator response at x and θ .
simulator run at limited input settings
 $\eta = (\eta(x_1^*, \theta_1^*), \dots, \eta(x_m^*, \theta_m^*))^T$
treat $\eta(\cdot, \cdot)$ as a random function
use GP prior for $\eta(\cdot, \cdot)$
 $y(x)$ experimental observation of the physical system
 $e(x)$ observation error of the experimental data

$$y(x) = \zeta(x) + e(x)$$

$$y(x) = \eta(x, \theta) + e(x)$$

Accounting for limited simulation runs



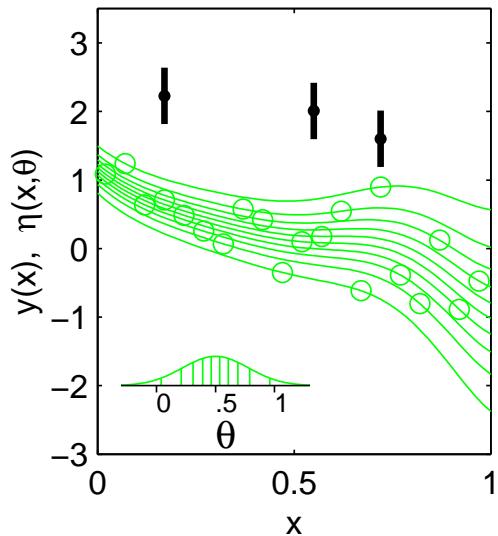
Again, standard Bayesian estimation gives:

$$\begin{aligned}\pi(\theta, \eta(\cdot, \cdot) | y(x)) \propto & L(y(x) | \eta(x, \theta)) \times \\ & \pi(\theta) \times \pi(\eta(\cdot, \cdot))\end{aligned}$$

- Posterior means and quantiles shown.
- Uncertainty in θ and $\eta(x, \theta)$ are incorporated into the forecast.
- Gaussian process models for $\eta(\cdot, \cdot)$.

Accounting for model discrepancy

prior uncertainty



- Borrows from Kennedy and O'Hagan (2001).

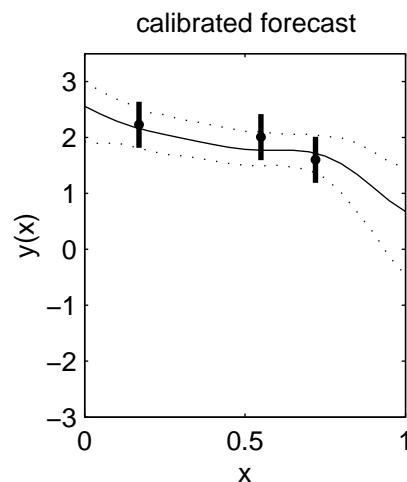
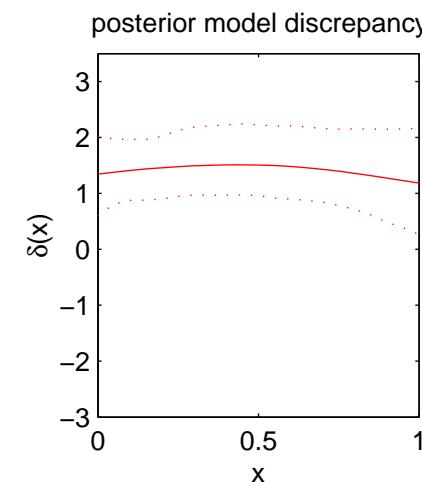
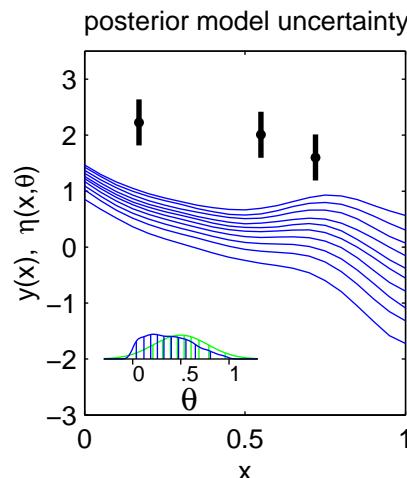
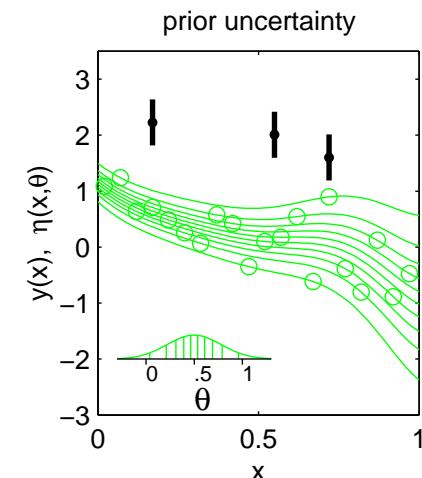
x	model or system inputs
θ	model or system inputs
$\zeta(x)$	true physical system response given inputs x
$\eta(x, \theta)$	simulator response at x and θ .
$y(x)$	experimental observation of the physical system
$\delta(x)$	discrepancy between $\zeta(x)$ and $\eta(x, \theta)$ may be decomposed into numerical error and bias
$e(x)$	observation error of the experimental data

$$y(x) = \zeta(x) + e(x)$$

$$y(x) = \eta(x, \theta) + \delta(x) + e(x)$$

$$y(x) = \eta(x, \theta) + \delta_n(x) + \delta_b(x) + e(x)$$

Accounting for model discrepancy



Again, standard Bayesian estimation gives:

$$\pi(\theta, \delta_n, \delta_b | y(x)) \propto L(y(x) | \eta(x, \theta), \delta(x)) \times \pi(\theta) \times \pi(\eta) \times \pi(\delta)$$

- Posterior means and 90% CI's shown.
- Posterior prediction for $\zeta(x)$ is obtained by computing the posterior distribution for $\eta(x, \theta) + \delta(x)$
- Uncertainty in θ , $\eta(x, t)$, and $\delta(x)$ are incorporated into the forecast.
- Gaussian process models for $\eta(x, t)$ and $\delta(x)$

Application: implosions of steel cylinders – Neddermeyer '43

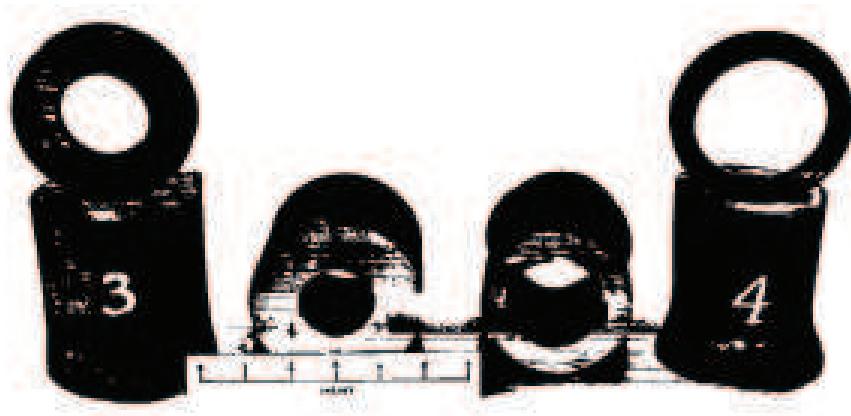
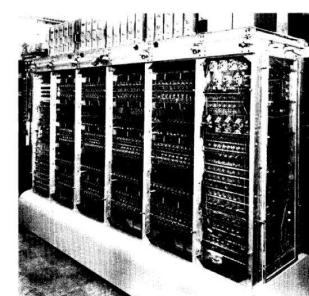
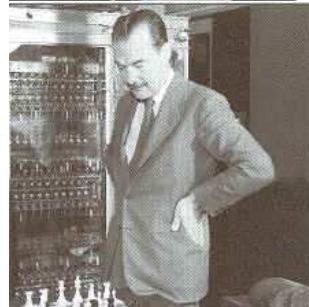
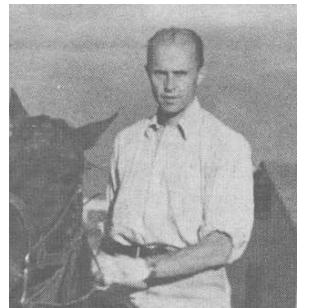


Fig. 13. Exp. 3: 4" OD, 1" wall, 8" long
TNT, 1" thick, $7\frac{1}{2}$ " long

Exp. 4: 4" OD, 1" wall, 8" long
TNT, 1" thick, $7\frac{1}{2}$ " long
Shearing ruptures are shown along lines where detonation waves meet.

- Initial work on implosion for fat man.
- Use high explosive (HE) to crush steel cylindrical shells
- Investigate the feasibility of a controlled implosion

Some History



Early work on cylinders called “beer can experiments.”

- Early work not encouraging:

“...I question Dr. Neddermeyer's seriousness...” – Deke Parsons.

“It stinks.” – R. Feynman

Teller and VonNeumann were quite supportive of the implosion idea

Data on collapsing cylinder from high speed photography.

Symmetrical implosion eventually accomplished using HE lenses by Kistiakowsky.

Implosion played a key role in early computer experiments.

Feynman worked on implosion calculations with IBM accounting machines.

Eventually first computer with addressable memory was developed (MANIAC 1).

The Experiments

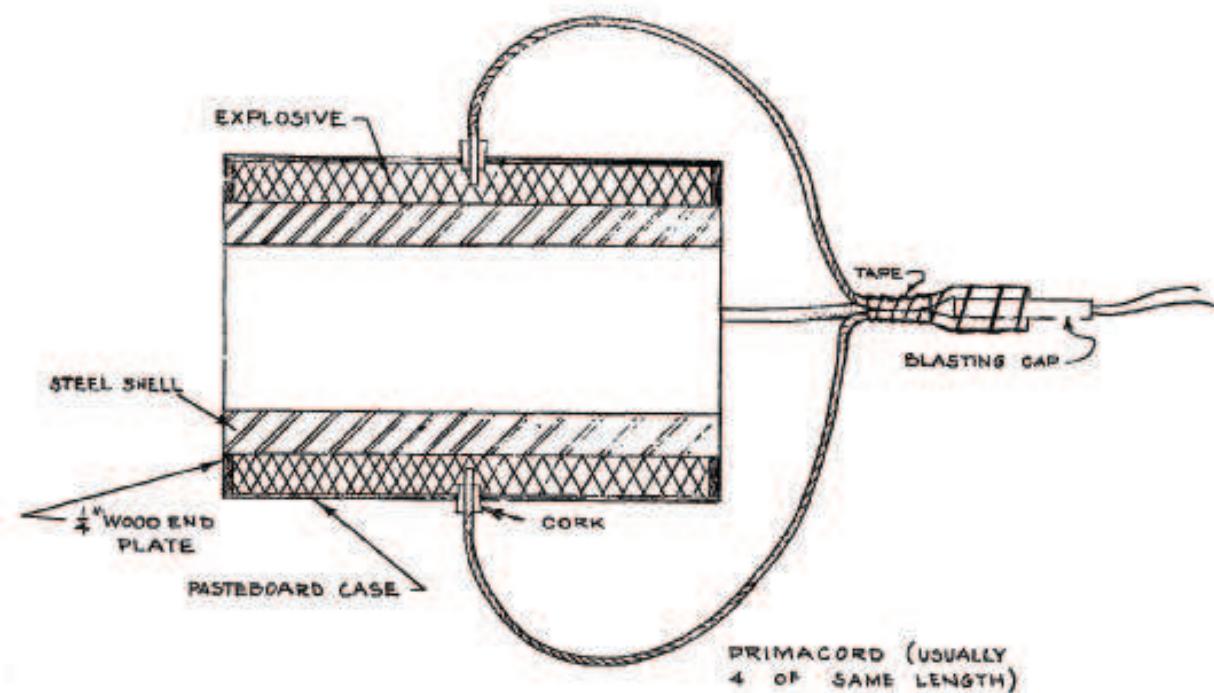


FIG. 10
SECTION OF TYPICAL ASSEMBLY
DRAWN TO SCALE OF EXPERIMENT # 26



Fig. 14. Exp. 9: 3" OD, $\frac{1}{2}$ " wall, 8" long
TNT, $1\frac{1}{2}$ " thick, $7\frac{1}{2}$ " long

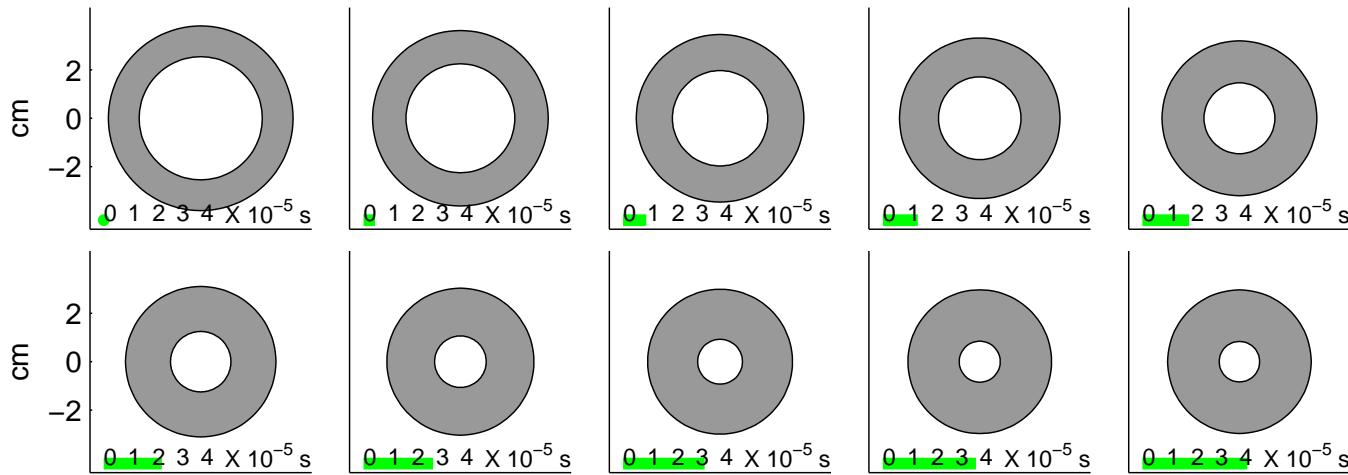
Exp. 11: 3" OD, $\frac{1}{2}$ " wall, 8" long, same charge

Both detonated from 4 points at lower end in photograph

Fig. 15. Exp. 13: 3" OD, $\frac{1}{2}$ " wall, 8" long
Comp. C, $1\frac{1}{2}$ " thick, $7\frac{1}{2}$ " long
Cf. Fig. 11, note uniform collapse when excessive charge is used

Exp. 14: 3" OD, $\frac{1}{2}$ " wall, 8" long
Comp. C, $1\frac{1}{2}$ " thick, $7\frac{1}{2}$ " long
Plastic flow can be seen through end of cylinder

Neddermeyer's Model



Energy from HE imparts an initial inward velocity to the cylinder

$$v_0 = \frac{m_e}{m} \sqrt{\frac{2u_0}{1 + m_e/m}}$$

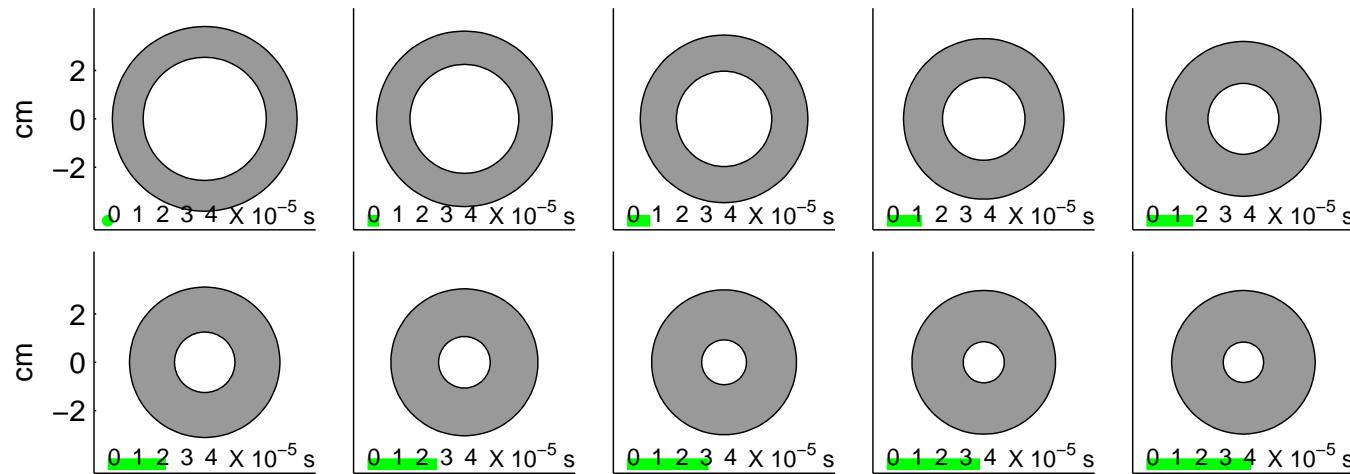
mass ratio m_e/m of HE to steel; u_0 energy per unit mass from HE.

Energy converts to work done on the cylinder:

$$\text{work per unit mass} = w = \frac{s}{2\rho(1-\lambda)} \left\{ r_i^2 \log r_i^2 - r_o^2 \log r_o^2 + \lambda^2 \log \lambda^2 \right\}$$

r_i = scaled inner radius; r_o = scaled outer radius; λ = initial r_i/r_o ; s = steel yielding stress; ρ = density of steel.

Neddermeyer's Model



ODE:
$$\frac{dr}{dt} = \left[\frac{1}{R_1^2 f(r)} \left\{ v_0^2 - \frac{s}{\rho} g(r) \right\} \right]^{\frac{1}{2}}$$

where

r = inner radius of cylinder – varies with time

R_1 = initial outer radius of cylinder

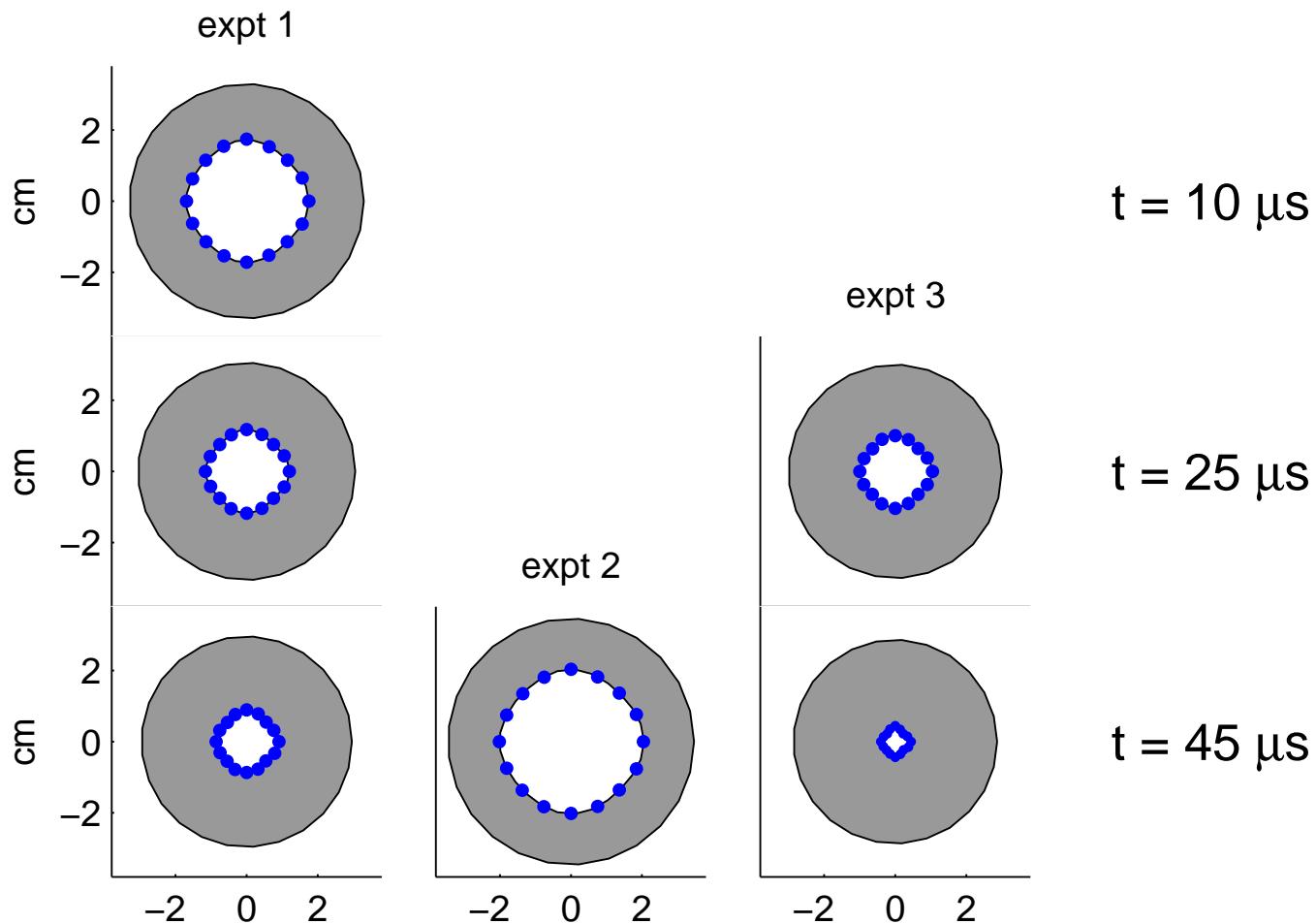
$$f(r) = \frac{r^2}{1 - \lambda^2} \ln \left(\frac{r^2 + 1 - \lambda^2}{r^2} \right)$$

$$g(r) = (1 - \lambda^2)^{-1} [r^2 \ln r^2 - (r^2 + 1 - \lambda^2) \ln(r^2 + 1 - \lambda^2) - \lambda^2 \ln \lambda^2]$$

λ = initial ratio of cylinder $r(t = 0)/R_1$

constant volume condition: $r_{\text{outer}}^2 - r^2 = 1 - \lambda^2$

Goal: use experimental data to calibrate s and u_0 ; obtain prediction uncertainty for new experiment



$$m_e/m \approx .32$$

$$m_e/m \approx .17$$

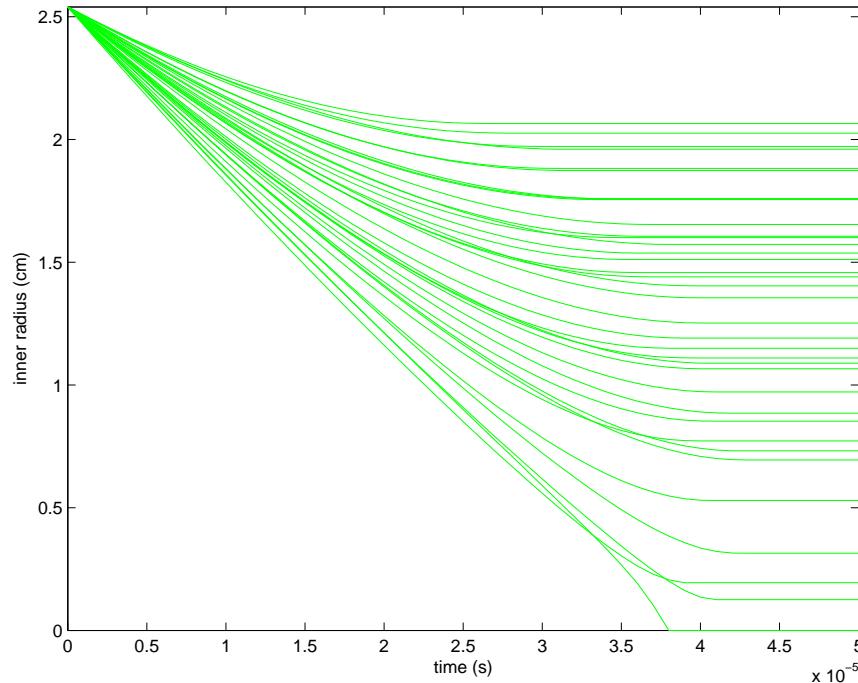
$$m_e/m \approx .36$$

Hypothetical data obtained from photos at different times during the 3 experimental implosions. All cylinders had a 1.5in outer and a 1.0in inner radius. ($\lambda = \frac{2}{3}$).

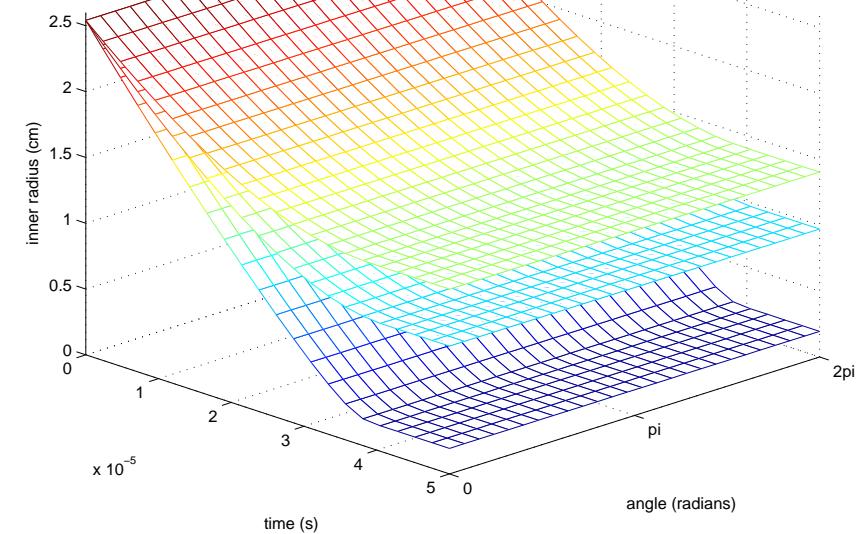
Carry out simulated implosions using Neddermeyer's model

Sequence of runs carried at m input settings $(x^*, \theta_1^*, \theta_2^*) = (m_e/m, s, u_0)$ varying over predefined ranges using an OA($32, 4^3$)-based LH design.

$$\begin{pmatrix} x_1^* & \theta_{11}^* & \theta_{12}^* \\ \vdots & \vdots & \vdots \\ x_m^* & \theta_{m1}^* & \theta_{m2}^* \end{pmatrix}$$



radius by time



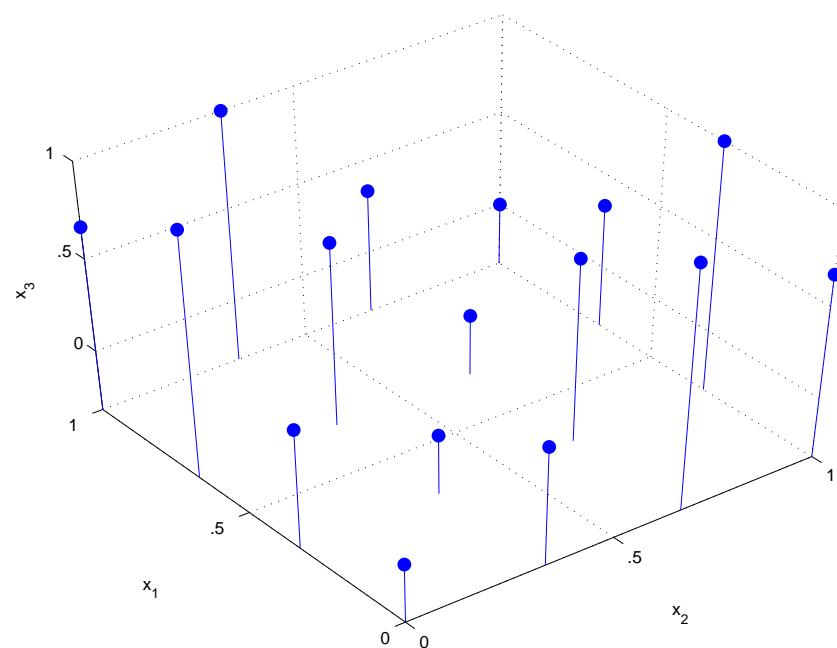
radius by time and angle ϕ .

Each simulation produces a $n_\eta = 22 \cdot 26$ vector of radii for 22 times \times 26 angles.

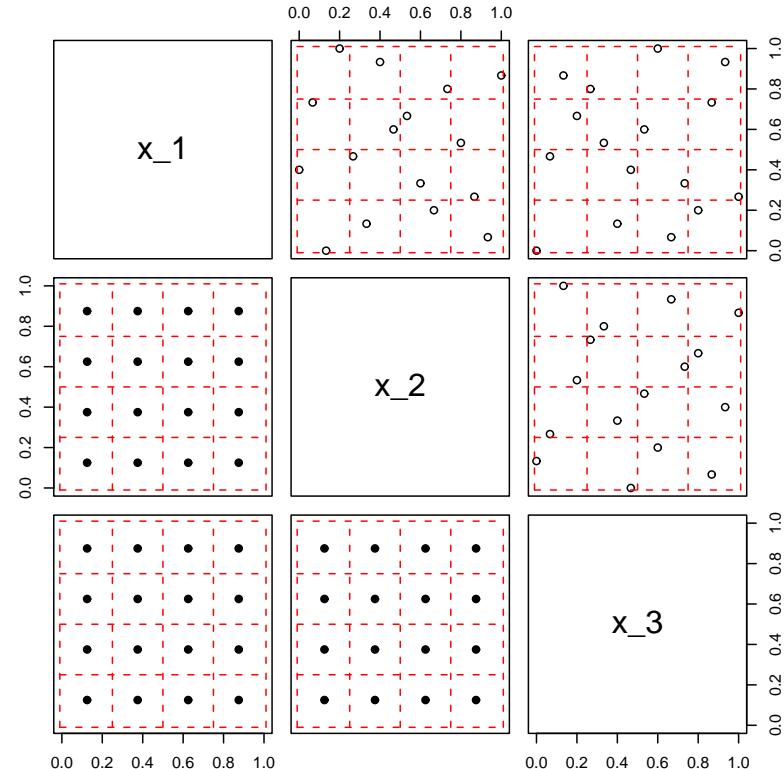
Generating OA-based LH designs

Example: $N = 16$, 3 factors each at 4 levels

OA(16, 4³) design



induced LH design



Ensures some higher dimensional filling relative to standard LH designs.

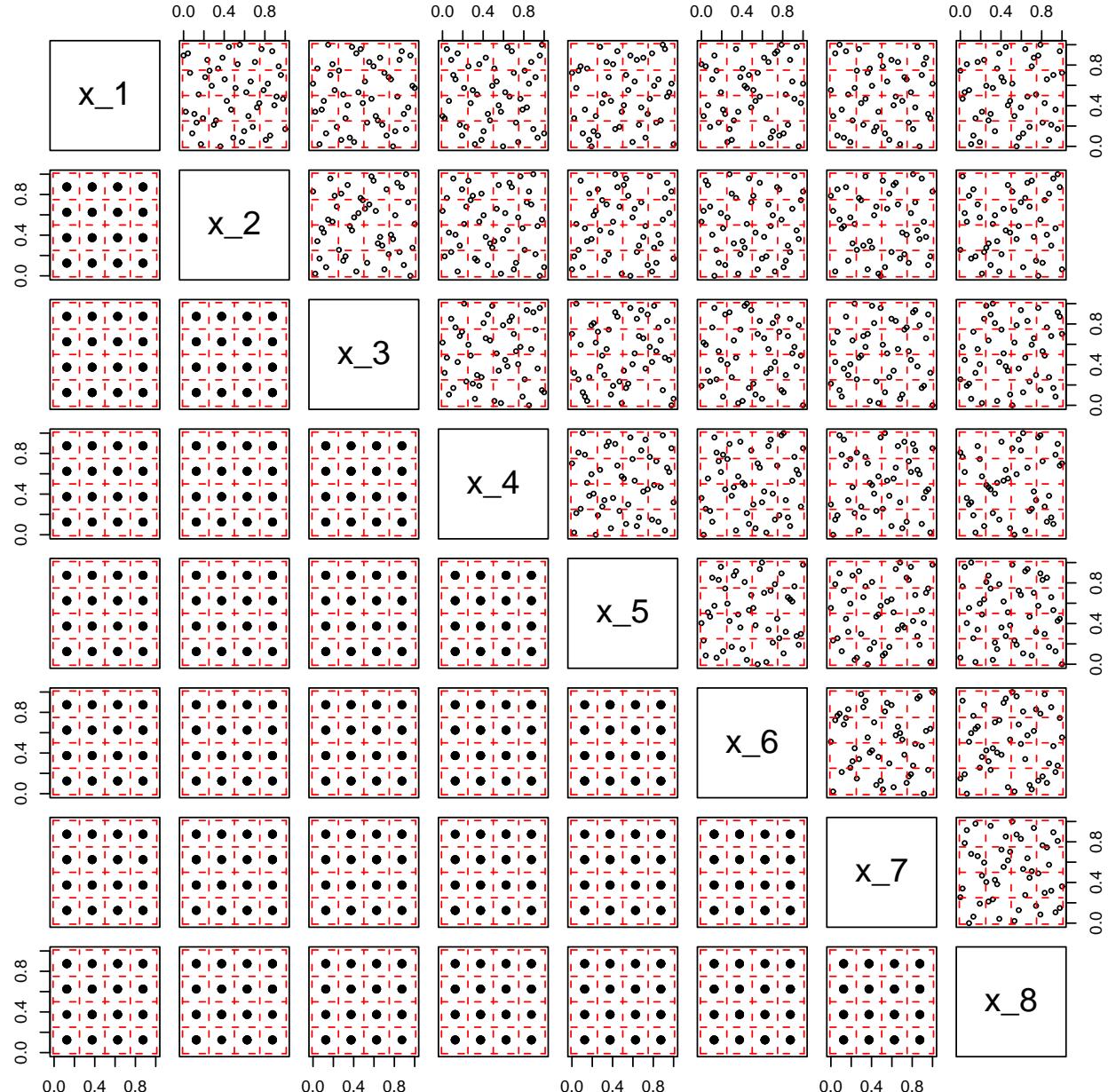
Generating (nearly) OA-based LH designs

Example:

NOA(48, 4⁸)

$N = 48$, 8 factors
each at 4 levels

columns of NOA
design matrix X
are not exactly or-
thogonal \Rightarrow al-
lows more factors
with good higher
dimensional prop-
erties.



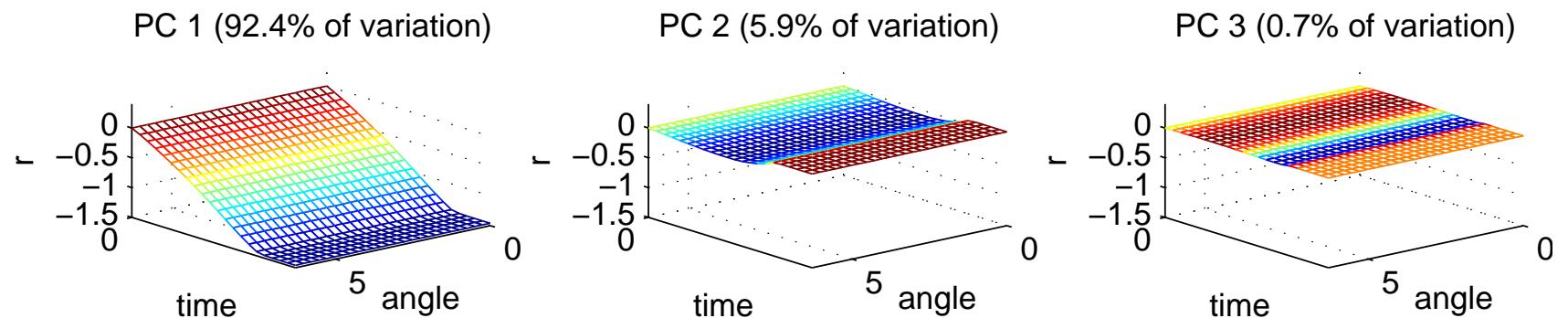
PC representation of simulation output

$\Xi = [\eta_1; \dots; \eta_m]$ – a $n_\eta \times m$ matrix that holds output of m simulations

SVD decomposition: $\Xi = UDV^T$

K_η is 1st p_η columns of $[\frac{1}{\sqrt{m}}UD]$ – columns of $[\sqrt{m}V^T]$ have variance 1

Cylinder example:



$p_\eta = 3$ PC's: $K_\eta = [k_1; k_2; k_3]$ – each vector k_i holds trace of PC i .

k_i 's do not change with ϕ – from symmetry of Neddermeyer's model.

Simulated trace $\eta(x_i^*, \theta_{i1}^*, \theta_{i2}^*) = K_\eta w(x_i^*, \theta_{i1}^*, \theta_{i2}^*) + \epsilon_i$, ϵ_i 's $\stackrel{iid}{\sim} N(0, \lambda_\eta^{-1})$, for any set of tried simulation inputs $(x_i^*, \theta_{i1}^*, \theta_{i2}^*)$.

Gaussian process models for PC weights

Want to evaluate $\eta(x, \theta_1, \theta_2)$ at arbitrary input setting (x, θ_1, θ_2) .

Also want analysis to account for uncertainty here.

Approach: model each PC weight as a Gaussian process:

$$w_i(x, \theta_1, \theta_2) \sim \text{GP}(0, \lambda_{wi}^{-1} R((x, \theta), (x', \theta'); \rho_{wi}))$$

where

$$R((x, \theta), (x', \theta'); \rho_{wi}) = \prod_{k=1}^{p_x} \rho_{wik}^{-4(x_k - x'_k)^2} \times \prod_{k=1}^{p_\theta} \rho_{wi(k+p_x)}^{-4(\theta_k - \theta'_k)^2} \quad (1)$$

Restricting to the design settings $\begin{pmatrix} x_1^* & \theta_{11}^* & \theta_{12}^* \\ \vdots & \vdots & \vdots \\ x_m^* & \theta_{m1}^* & \theta_{m2}^* \end{pmatrix}$ and specifying

$$w_i = (w_i(x_1^*, \theta_{11}^*, \theta_{12}^*), \dots, w_i(x_m^*, \theta_{m1}^*, \theta_{m2}^*))^T$$

gives

$$w_i \stackrel{iid}{\sim} N(0, \lambda_{wi}^{-1} R((x^*, \theta^*); \rho_{wi})), \quad i = 1, \dots, p_\eta$$

where $R((x^*, \theta^*); \rho_{wi})_{m \times m}$ is given by (1).

note: additional nugget term $w_i \stackrel{iid}{\sim} N(0, \lambda_{wi}^{-1} R((x^, \theta^*); \rho_{wi}) + \lambda_{\epsilon i}^{-1} I_m)$, $i = 1, \dots, p_\eta$, may be useful.

Gaussian process models for PC weights

At the m simulation input settings the mp_η -vector w has prior distribution

$$w = \begin{pmatrix} w_1 \\ \vdots \\ w_{p_\eta} \end{pmatrix} \sim N \left(\begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} \lambda_{w1}^{-1} R((x^*, \theta^*); \rho_{w1}) & 0 & \dots & 0 \\ 0 & \ddots & & 0 \\ \vdots & & 0 & \lambda_{wp_\eta}^{-1} R((x^*, \theta^*); \rho_{wp_\eta}) \\ 0 & & \dots & 0 \end{pmatrix} \right)$$

$$\Rightarrow w \sim N(0, \Sigma_w);$$

note $\Sigma_w = I_{p_\eta} \otimes \lambda_w^{-1} R((x^*, \theta^*); \rho_w)$ can break down.

Emulator likelihood: $\eta = \text{vec}([\eta(x_1^*, \theta_{11}^*, \theta_{12}^*); \dots; \eta(x_m^*, \theta_{m1}^*, \theta_{m2}^*)])$

$$L(\eta|w, \lambda_\eta) \propto \lambda_\eta^{\frac{mn_\eta}{2}} \exp \left\{ -\frac{1}{2} \lambda_\eta (\eta - Kw)^T (\eta - Kw) \right\}, \quad \lambda_\eta \sim \Gamma(a_\eta, b_\eta)$$

where n_η is the number of observations in a simulated trace and

$$\text{Equivalently} \quad K = [I_m \otimes k_1; \dots; I_m \otimes k_{p_\eta}].$$

$$\begin{aligned} L(\eta|w, \lambda_\eta) &\propto \lambda_\eta^{\frac{mp_\eta}{2}} \exp \left\{ -\frac{1}{2} \lambda_\eta (w - \hat{w})^T (K^T K) (w - \hat{w}) \right\} \times \\ &\quad \lambda_\eta^{\frac{m(n_\eta - p_\eta)}{2}} \exp \left\{ -\frac{1}{2} \lambda_\eta \eta^T (I - K(K^T K)^{-1} K^T) \eta \right\} \\ &\propto \lambda_\eta^{\frac{mp_\eta}{2}} \exp \left\{ -\frac{1}{2} \lambda_\eta (w - \hat{w})^T (K^T K) (w - \hat{w}) \right\}, \quad \lambda_\eta \sim \Gamma(a'_\eta, b'_\eta) \end{aligned}$$

$$a'_\eta = a_\eta + \frac{m(n_\eta - p_\eta)}{2}, \quad b'_\eta = b_\eta + \frac{1}{2} \eta^T (I - K(K^T K)^{-1} K^T) \eta, \quad \hat{w} = (K^T K)^{-1} K^T \eta.$$

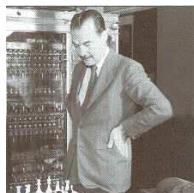
Gaussian process models for PC weights

Resulting posterior can then be based on computed PC weights \hat{w} :

$$\begin{aligned}\hat{w}|w, \lambda_\eta &\sim N(w, (\lambda_\eta K^T K)^{-1}) \\ w|\lambda_w, \rho_w &\sim N(0, \Sigma_w) \\ \Rightarrow \hat{w}|\lambda_\eta, \lambda_w, \rho_w &\sim N(0, (\lambda_\eta K^T K)^{-1} + \Sigma_w)\end{aligned}$$

Resulting posterior is then:

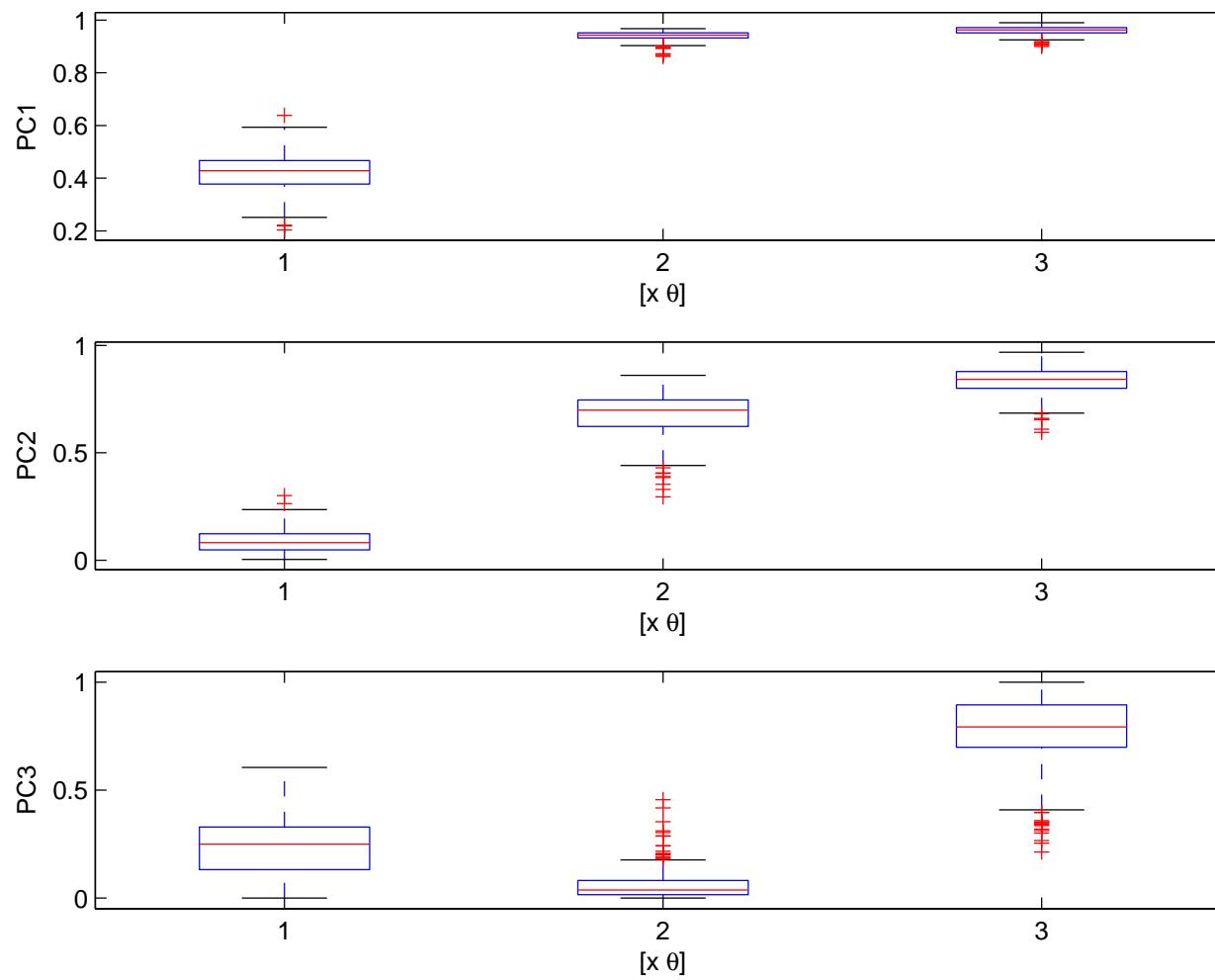
$$\begin{aligned}\pi(\lambda_\eta, \lambda_w, \rho_w | \hat{w}) \propto & \left|(\lambda_\eta K^T K)^{-1} + \Sigma_w\right|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\hat{w}^T([\lambda_\eta K^T K]^{-1} + \Sigma_w)^{-1}\hat{w}\right\} \times \\ & \lambda_\eta^{a'_\eta - 1} e^{-b'_\eta \lambda_\eta} \times \prod_{i=1}^{p_\eta} \lambda_{wi}^{a_w - 1} e^{-b_w \lambda_{wi}} \times \\ & \prod_{i=1}^{p_\eta} \left\{ \prod_{j=1}^{p_x} (1 - \rho_{wij})^{b_\rho - 1} \prod_{j=1}^{p_\theta} (1 - \rho_{wi(j+p_x)})^{b_\rho - 1} \right\}\end{aligned}$$



MCMC via Metropolis works fine here.

Bounded range of ρ_{wij} 's facilitates MCMC.

Posterior distribution of ρ_w



Separate models by PC

More opportunity to take advantage of effect sparsity

Predicting simulator output at untried $(x^*, \theta_1^*, \theta_2^*)$

Want $\eta(x^*, \theta_1^*, \theta_2^*) = Kw(x^*, \theta_1^*, \theta_2^*)$

For a given draw $(\lambda_\eta, \lambda_w, \rho_w)$ a draw of w^* can be produced:

$$\begin{pmatrix} \hat{w} \\ w^* \end{pmatrix} \sim N \left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \left[\begin{pmatrix} (\lambda_\eta K^T K)^{-1} & 0 \\ 0 & 0 \end{pmatrix} + \Sigma_{w,w^*}(\lambda_w, \rho_w) \right] \right)$$

Define

$$V = \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix} = \left[\begin{pmatrix} (\lambda_\eta K^T K)^{-1} & 0 \\ 0 & 0 \end{pmatrix} + \Sigma_{w,w^*}(\lambda_w, \rho_w) \right]$$

Then

$$w^* | \hat{w} \sim N(V_{21}V_{11}^{-1}\hat{w}, V_{22} - V_{21}V_{11}^{-1}V_{12})$$

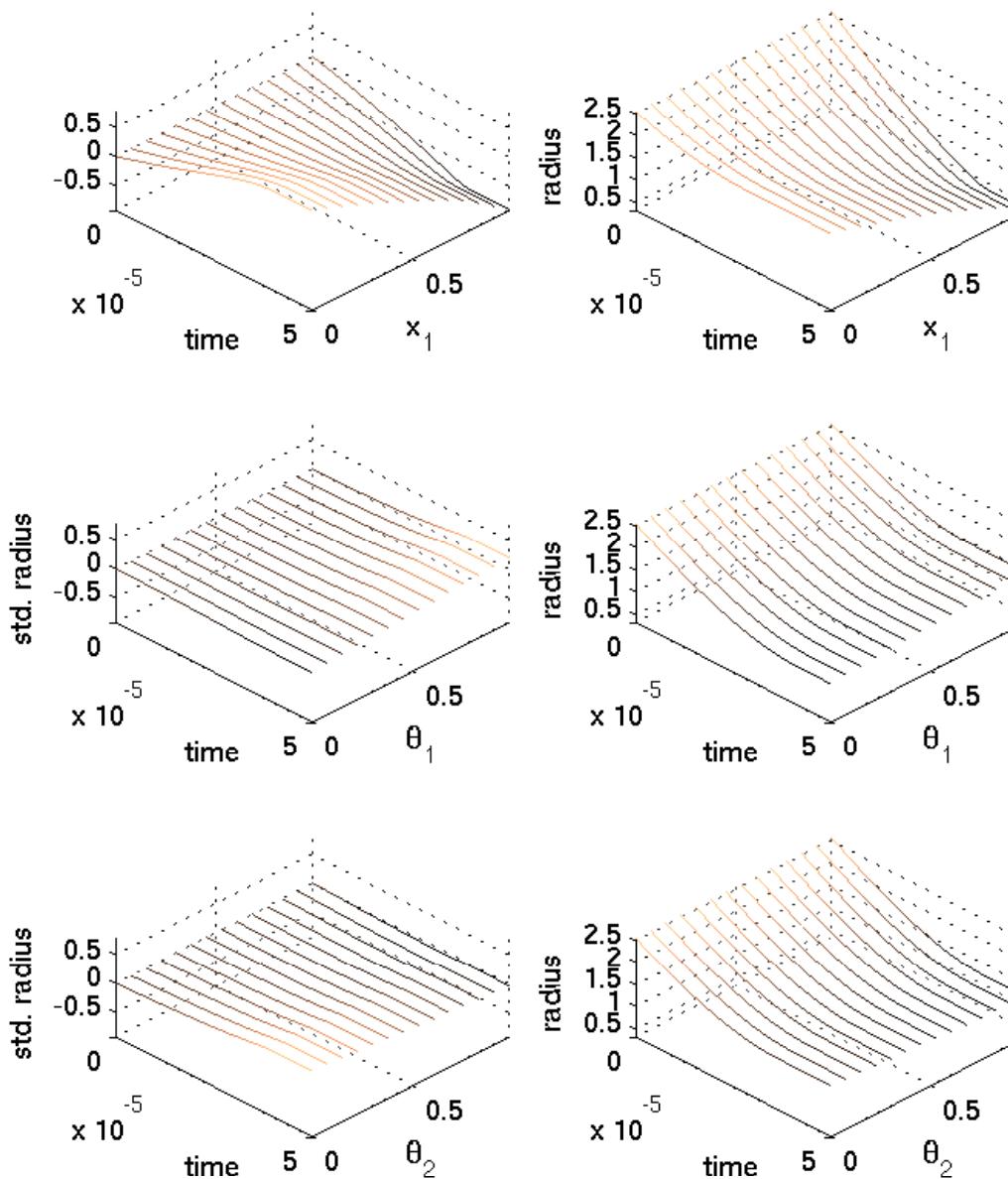
Realizations can be generated from sample of MCMC output.

Lots of info (data?) makes conditioning on point estimate $(\hat{\lambda}_\eta, \hat{\lambda}_w, \hat{\rho}_w)$ a good approximation to the posterior.

Posterior mean or median work well for $(\hat{\lambda}_\eta, \hat{\lambda}_w, \hat{\rho}_w)$

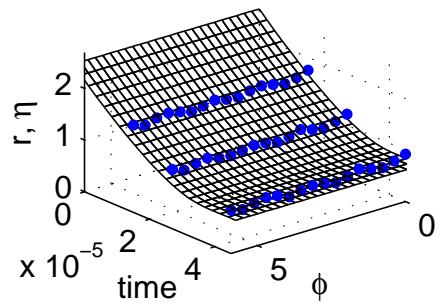
Exploring sensitivity of simulator output to model inputs

Simulator predictions varying 1 input, holding others at nominal



Basic formulation – borrows from Kennedy and O'Hagan (2001)

Experiment 1



(t, ϕ)

simulation output space

x

experimental conditions

θ

calibration parameters

$\zeta(x)$

true physical system response given conditions x

$\eta(x, \theta)$

simulator response at x and θ .

$y(x)$

experimental observation of the physical system

$\delta(x)$

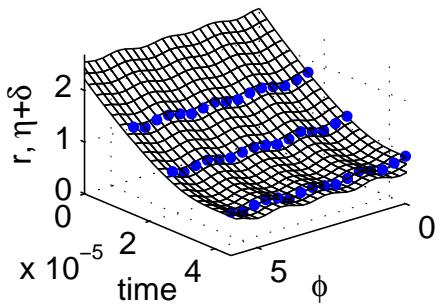
discrepancy between $\zeta(x)$ and $\eta(x, \theta)$

may be decomposed into numerical error and bias

observation error of the experimental data

$$y(x) = \zeta(x) + e(x)$$

$$y(x) = \eta(x, \theta) + \delta(x) + e(x)$$



$e(x)$

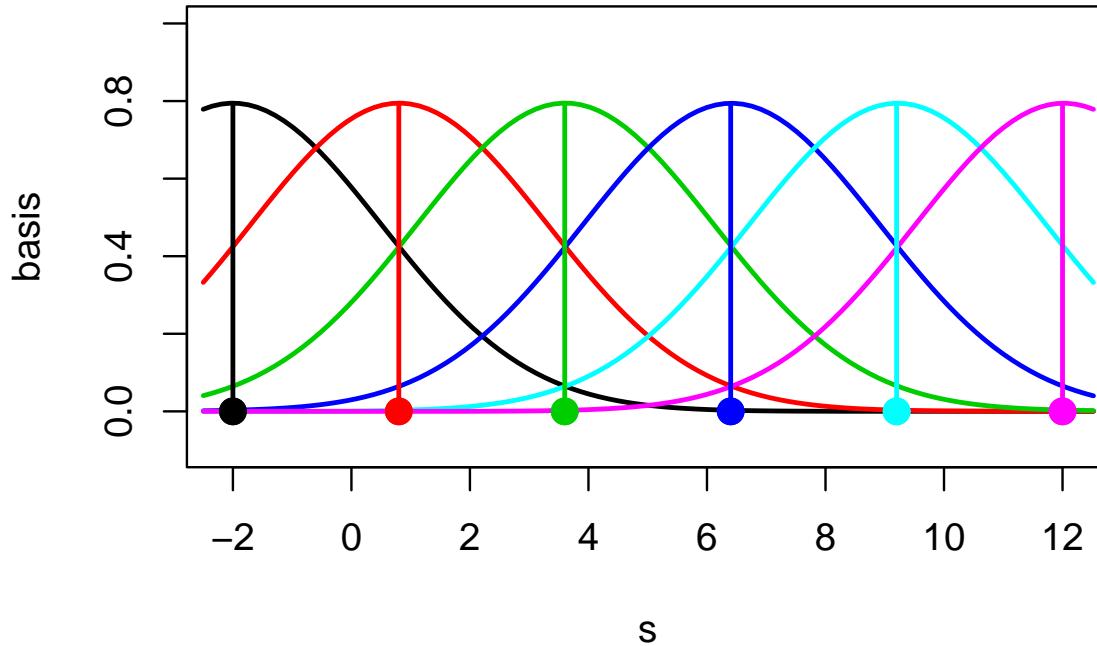
$$x = m_e/m \approx .32$$

$$\theta_1 = s \approx ?$$

$$\theta_2 = u_0 \approx ?$$

Kernel basis representation for spatial processes $\delta(s)$

Define p_δ basis functions $d_1(s), \dots, d_{p_\delta}(s)$.



Here $d_j(s)$ is normal density cetered at spatial location ω_j :

$$d_j(s) = \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2}(s - \omega_j)^2\right\}$$

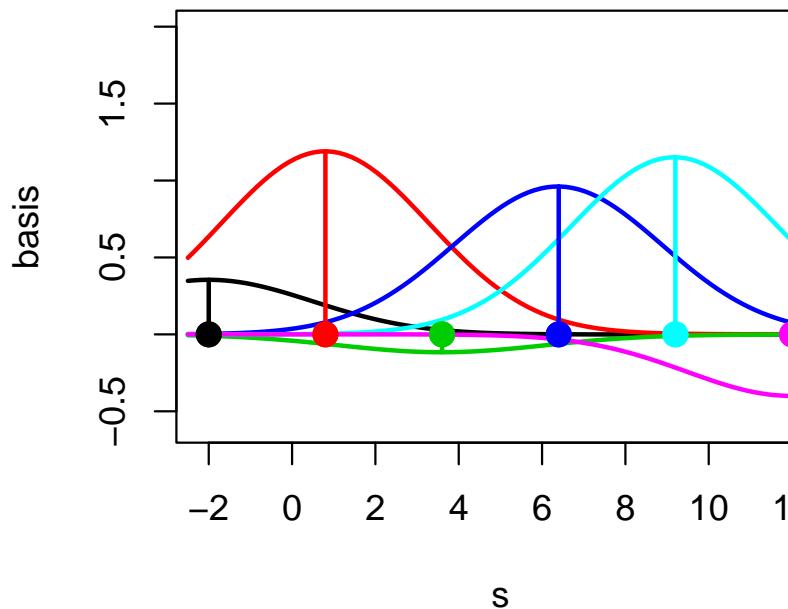
$$\text{set } \delta(s) = \sum_{j=1}^{p_\delta} d_j(s) v_j \text{ where } v \sim N(0, \lambda_v^{-1} I_{p_\delta}).$$

Can represent $\delta = (\delta(s_1), \dots, \delta(s_n))^T$ as $\delta = Dv$ where

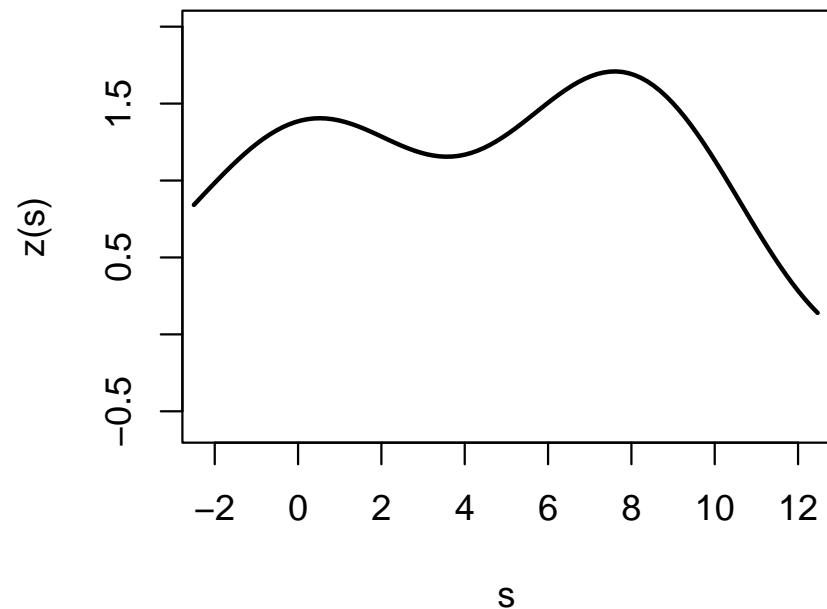
$$D_{ij} = d_j(s_i)$$

v and $d(s)$ determine spatial processes $\delta(s)$

$$d_j(s)v_j$$



$$\delta(s)$$

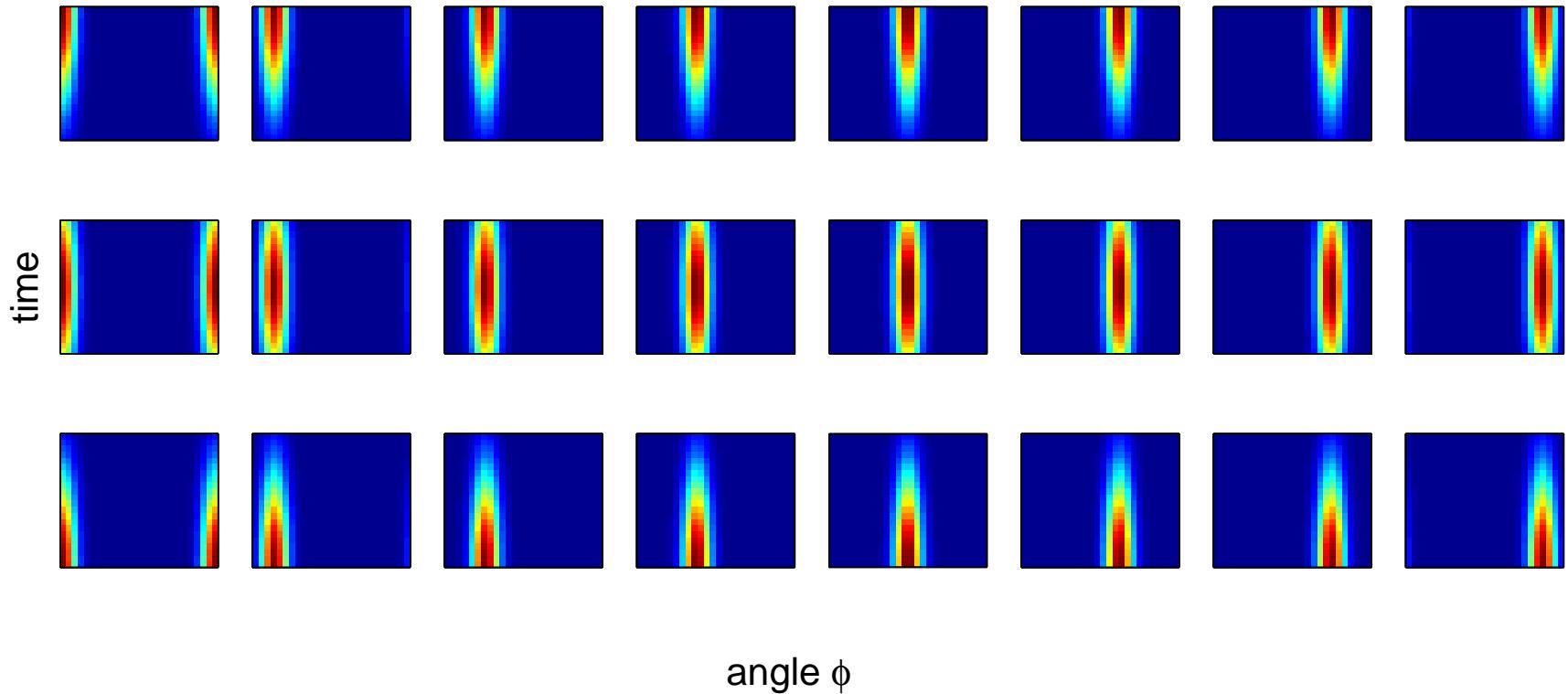


Continuous representation:

$$\delta(s) = \sum_{j=1}^{p_\delta} d_j(s)v_j \text{ where } v \sim N(0, \lambda_v^{-1} I_{p_\delta}).$$

Discrete representation: For $\delta = (\delta(s_1), \dots, \delta(s_n))^T$, $\delta = Dv$ where $D_{ij} = d_j(s_i)$

Basis representation of discrepancy



Represent discrepancy $\delta(x)$ using basis functions and weights

$p_\delta = 24$ basis functions over (t, ϕ) ; $D = [d_1; \dots; d_{p_\delta}]$; d_k 's hold basis.

$$\delta(x) = Dv(x) \text{ where } v(x) \sim \text{GP} \left(0, \lambda_v^{-1} I_{p_\delta} \otimes R(x, x'; \rho_v) \right)$$

with

$$R(x, x'; \rho_v) = \prod_{k=1}^{p_x} \rho_{vk}^{-4(x_k - x'_k)^2} \quad (2)$$

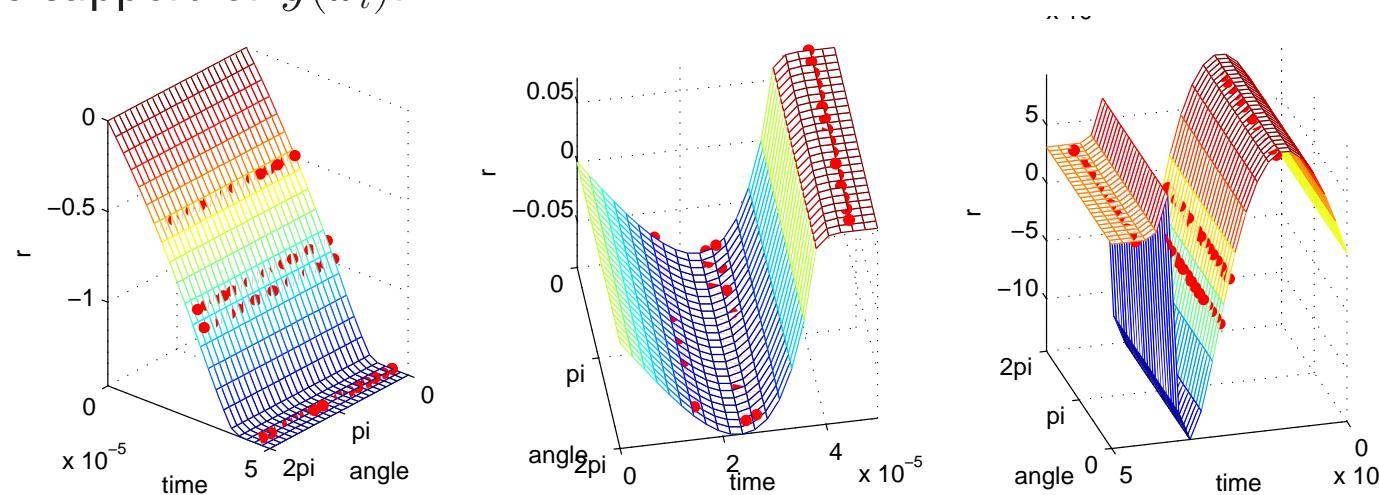
Integrated model formulation

Data $y(x_1), \dots, y(x_n)$ collected for n experiments at input conditions x_1, \dots, x_n .

Each $y(x_i)$ is a collection of n_{y_i} measurements over points indexed by (t, ϕ) .

$$\begin{aligned} y(x_i) &= \eta(x_i, \theta) + \delta(x_i) + e_i \\ &= K_i w(x_i, \theta) + D_i v(x_i) + e_i \\ y(x_i)|w(x_i, \theta), v(x_i), \lambda_y &\sim N \left([D_i; K_i] \begin{pmatrix} v(x_i) \\ w(x_i, \theta) \end{pmatrix}, (\lambda_y W_i)^{-1} \right) \end{aligned}$$

Since support of each $y(x_i)$ varies and doesn't match that of sims, the basis vectors in K_i must be interpolated from K_η ; similarly, D_i must be computed from the support of $y(x_i)$:



*note: cubic spline interpolation over (time, ϕ) used here.

Integrated model formulation

Define

$n_y = n_{y_1} + \dots + n_{y_n}$, the total number of experimental data points,

y to be the n_y -vector from concatenation of the $y(x_i)$'s,

$v = \text{vec}([v(x_1); \dots; v(x_n)]^T)$ and

$u(\theta) = \text{vec}([w(x_1, \theta_1, \theta_2); \dots; w(x_n, \theta_1, \theta_2)]^T)$

$$y|v, u(\theta), \lambda_y \sim \mathbf{N}\left(B \begin{pmatrix} v \\ u(\theta) \end{pmatrix}, (\lambda_y W_y)^{-1}\right), \quad \lambda_y \sim \Gamma(a_y, b_y) \quad (3)$$

where

$W_y = \text{diag}(W_1, \dots, W_n)$ and

$$B = \text{diag}(D_1, \dots, D_n, K_1, \dots, K_n) \begin{pmatrix} P_D^T & 0 \\ 0 & P_K^T \end{pmatrix}$$

P_D and P_K are permutation matrices whose rows are given by:

$$P_D(j + n(i-1); \cdot) = e_{(j-1)p_\delta+i}^T, \quad i = 1, \dots, p_\delta; \quad j = 1, \dots, n$$

$$P_K(j + n(i-1); \cdot) = e_{(j-1)p_\eta+i}^T, \quad i = 1, \dots, p_\eta; \quad j = 1, \dots, n$$

Integrated model formulation (continued)

Equivalently (3) can be represented

$$\begin{pmatrix} \hat{v} \\ \hat{u} \end{pmatrix} \left| \begin{pmatrix} v \\ u(\theta) \end{pmatrix}, \lambda_y \sim \mathcal{N} \left(\begin{pmatrix} v \\ u(\theta) \end{pmatrix}, (\lambda_y B^T W_y B)^{-1} \right), \lambda_y \sim \Gamma(a'_y, b'_y) \right.$$

with

$$\begin{aligned} n_y &= n_{y_1} + \cdots + n_{y_n}, \quad \text{the total number of experimental data points} \\ \begin{pmatrix} \hat{v} \\ \hat{u} \end{pmatrix} &= (B^T W_y B)^{-1} B^T W_y y \\ a'_y &= a_y + \frac{1}{2}[n_y - n(p_\delta + p_\eta)] \\ b'_y &= b_y + \frac{1}{2} \left[\left(y - B \begin{pmatrix} \hat{v} \\ \hat{u} \end{pmatrix} \right)^T W_y \left(y - B \begin{pmatrix} \hat{v} \\ \hat{u} \end{pmatrix} \right) \right] \end{aligned}$$

dimension reduction

model simulator data and discrep

standard	$n_\eta \cdot m$	n_y
basis	$p_\eta \cdot m$	$n \cdot (p_\delta + p_\eta)$

Basis approach particularly efficient when n_η and n_y are large.

Marginal likelihood

The (marginal) likelihood $L(\hat{v}, \hat{u}, \hat{w} | \lambda_\eta, \lambda_w, \rho_w, \lambda_y, \lambda_v, \rho_v, \theta)$ has the form

$$\begin{pmatrix} \hat{v} \\ \hat{u} \\ \hat{w} \end{pmatrix} \sim N \left(\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \Lambda_y^{-1} & 0 & 0 \\ 0 & 0 & \Lambda_\eta^{-1} \\ 0 & \Lambda_\eta^{-1} & 0 \end{pmatrix} + \begin{pmatrix} \Sigma_v & 0 & 0 \\ 0 & \Sigma_{uw} & 0 \\ 0 & 0 & \Sigma_{uw} \end{pmatrix} \right)$$

where

$$\Lambda_y = \lambda_y B^T W_y B$$

$$\Lambda_\eta = \lambda_\eta K^T K$$

$$\Sigma_v = \lambda_v^{-1} I_{p_\eta} \otimes R(x, x; \rho_v)$$

$R(x, x; \rho_v)$ = $n \times n$ correlation matrix from applying (2) to the conditions x_1, \dots, x_n corresponding to the n experiments.

$$\Sigma_{uw} = \begin{pmatrix} \lambda_{w1}^{-1} R((x, \theta), (x, \theta); \rho_{w1}) & 0 & 0 & \lambda_{w1}^{-1} R((x, \theta), (x^*, \theta^*); \rho_{w1}) & 0 & 0 \\ 0 & \ddots & 0 & 0 & \ddots & 0 \\ 0 & 0 & \lambda_{wp_\eta}^{-1} R((x, \theta), (x, \theta); \rho_{wp_\eta}) & 0 & 0 & \lambda_{wp_\eta}^{-1} R((x, \theta), (x^*, \theta^*); \rho_{wp_\eta}) \\ \lambda_{w1}^{-1} R((x^*, \theta^*), (x, \theta); \rho_{w1}) & 0 & 0 & \lambda_{w1}^{-1} R((x^*, \theta^*), (x^*, \theta^*); \rho_{w1}) & 0 & 0 \\ 0 & \ddots & 0 & 0 & \ddots & 0 \\ 0 & 0 & \lambda_{wp_\eta}^{-1} R((x^*, \theta^*), (x, \theta); \rho_{wp_\eta}) & 0 & 0 & \lambda_{wp_\eta}^{-1} R((x^*, \theta^*), (x^*, \theta^*); \rho_{wp_\eta}) \end{pmatrix}$$

Permutation of Σ_{uw} is block diagonal \Rightarrow can speed up computations.

Only off diagonal blocks of Σ_{uw} depend on θ .

Posterior distribution

Likelihood: $L(\hat{v}, \hat{u}, \hat{w} | \lambda_\eta, \lambda_w, \rho_w, \lambda_y, \lambda_v, \rho_v, \theta)$

Prior: $\pi(\lambda_\eta, \lambda_w, \rho_w, \lambda_y, \lambda_v, \rho_v, \theta)$

⇒ Posterior:

$$\pi(\lambda_\eta, \lambda_w, \rho_w, \lambda_y, \lambda_v, \rho_v, \theta | \hat{v}, \hat{u}, \hat{w}) \propto L(\hat{v}, \hat{u}, \hat{w} | \lambda_\eta, \lambda_w, \rho_w, \lambda_y, \lambda_v, \rho_v, \theta) \times \pi(\lambda_\eta, \lambda_w, \rho_w, \lambda_y, \lambda_v, \rho_v, \theta)$$

Posterior exploration via MCMC

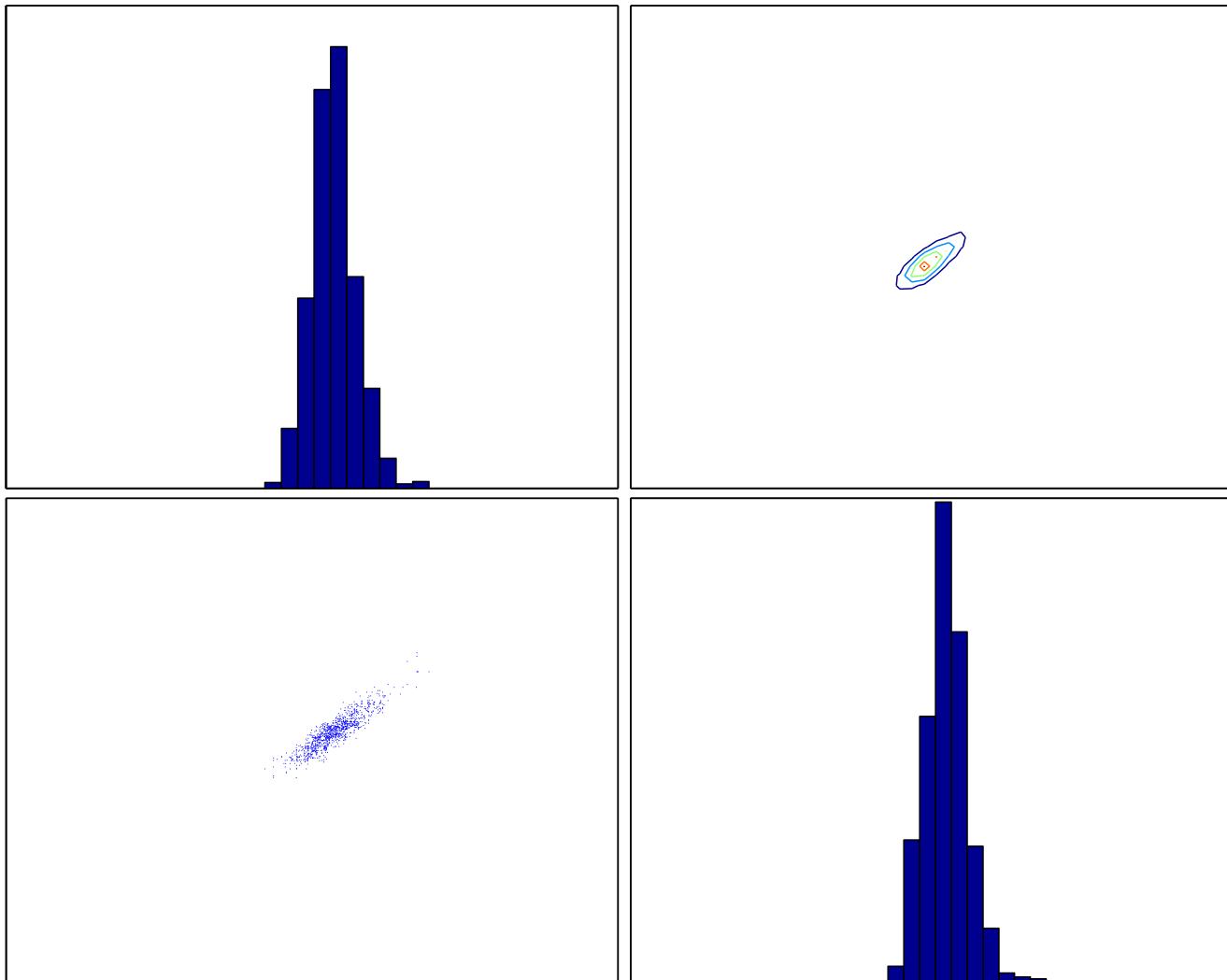
Can take advantage of structure and sparsity to speed up sampling.

A useful approximation to speed up posterior evaluation:

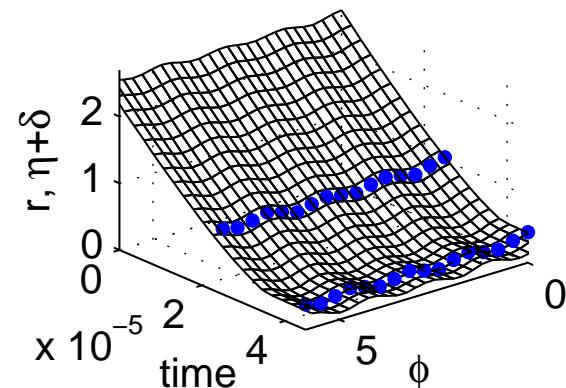
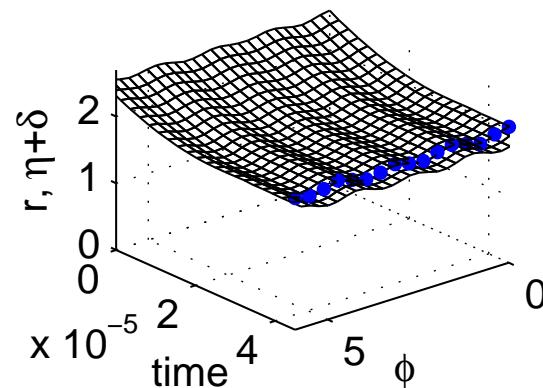
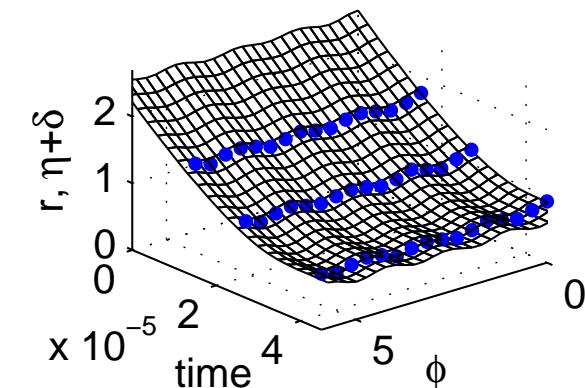
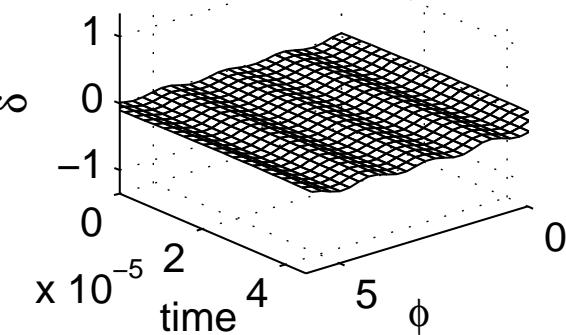
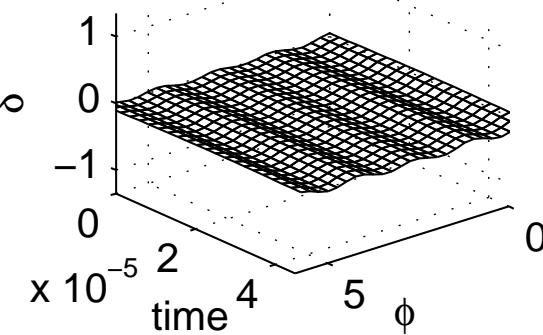
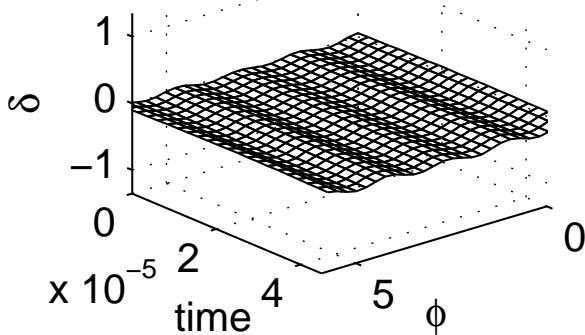
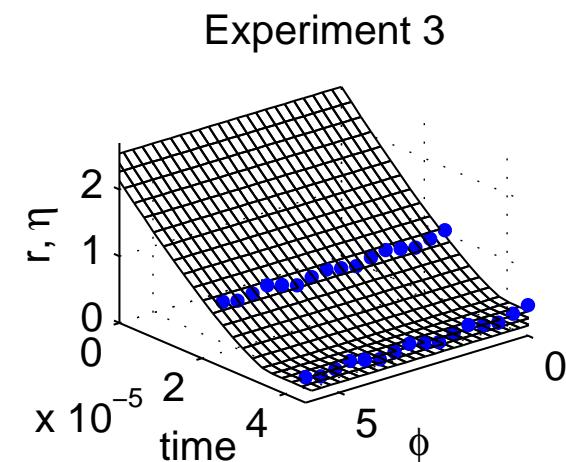
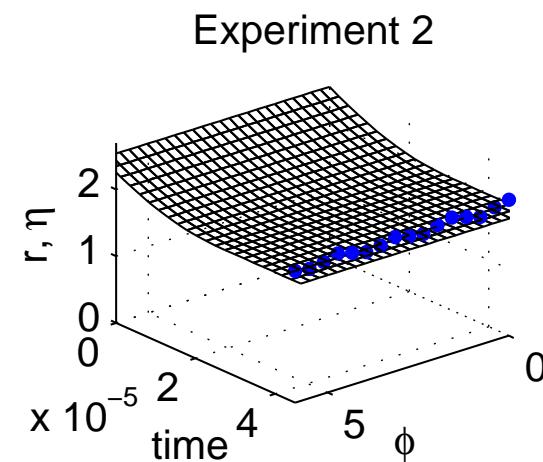
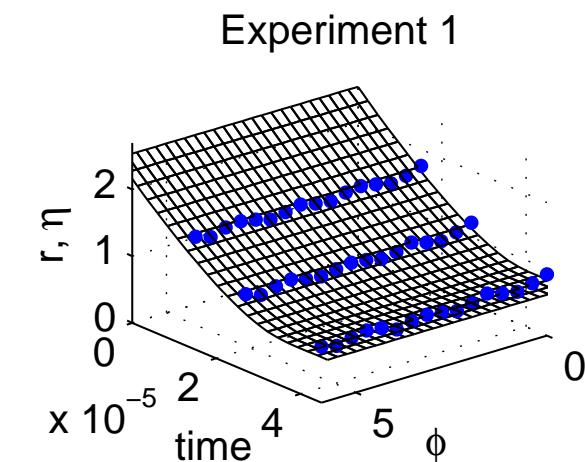
$$\begin{aligned} & \pi(\lambda_\eta, \lambda_w, \rho_w, \lambda_y, \lambda_v, \rho_v, \theta | \hat{v}, \hat{u}, \hat{w}) \\ & \propto L(\hat{w} | \lambda_\eta, \lambda_w, \rho_w) \times \pi(\lambda_\eta, \lambda_w, \rho_w) \times \\ & \quad L(\hat{v}, \hat{u} | \lambda_\eta, \lambda_w, \rho_w, \lambda_y, \lambda_v, \rho_v, \theta) \times \pi(\lambda_y, \lambda_v, \rho_v, \theta) \end{aligned}$$

In this approximation, experimental data is not used to inform about parameters $\lambda_\eta, \lambda_w, \rho_w$ which govern the simulator process $\eta(x, \theta)$.

Posterior distribution of model parameters (θ_1, θ_2)

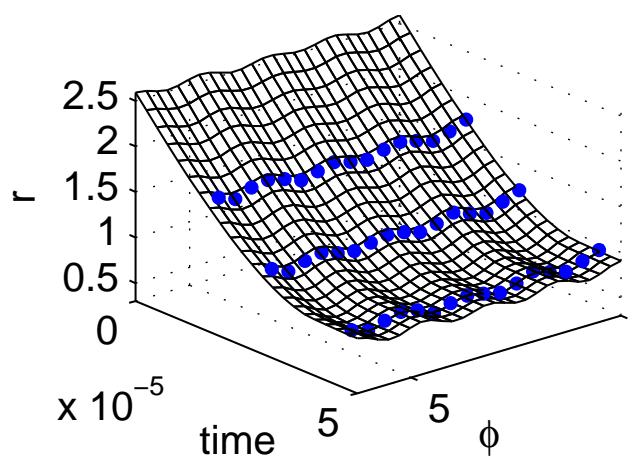


Posterior mean decomposition for each experiment

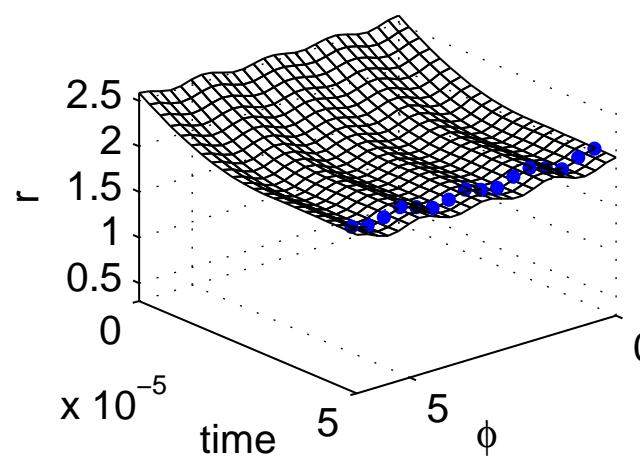


Posterior prediction for implosion in each experiment

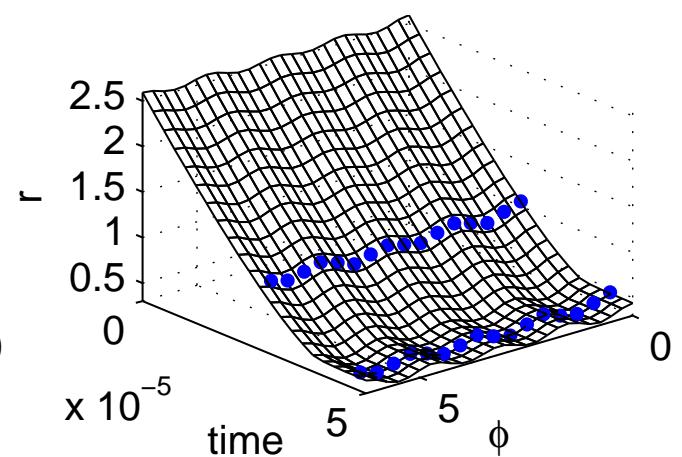
Experiment 1



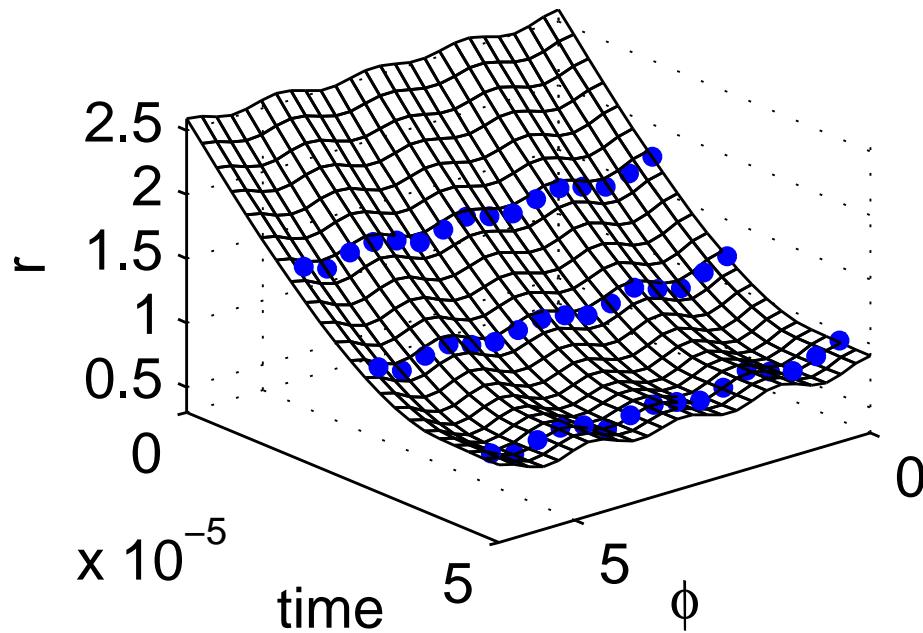
Experiment 2



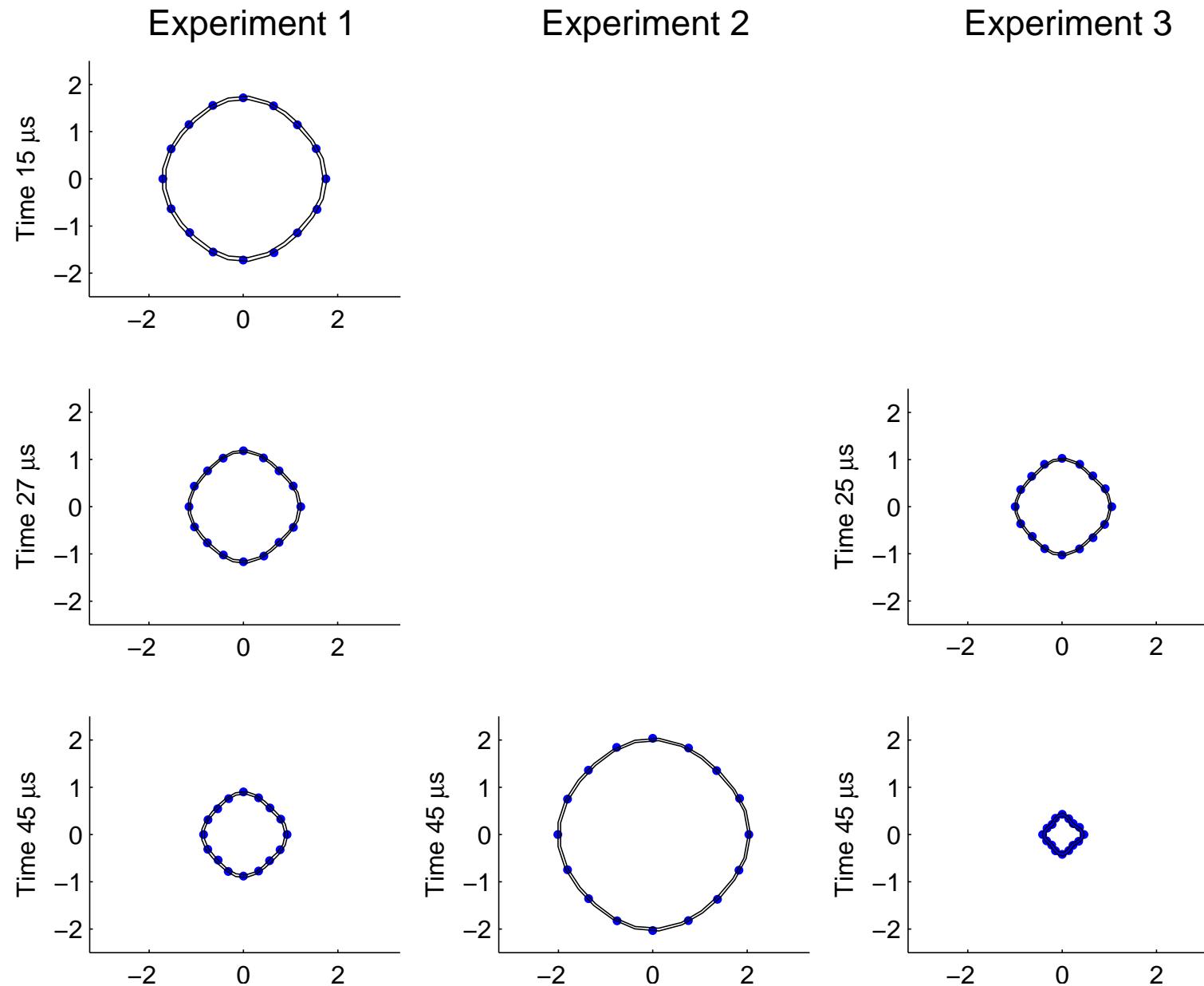
Experiment 3



Experiment 1



90% prediction intervals for implosions at exposure times

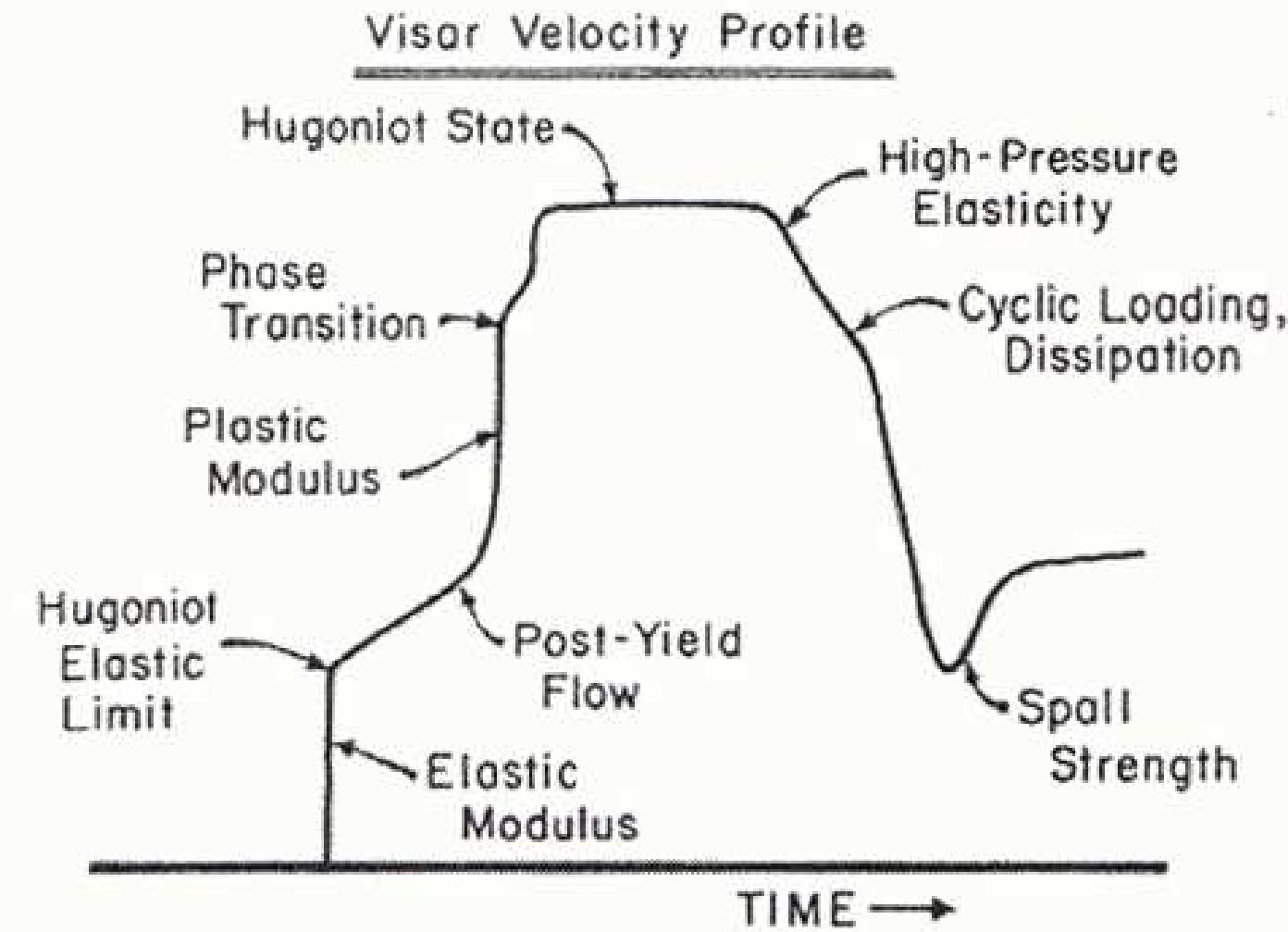


Predictions from separate analyses which hold data from the experiment being predicted.

Quantifying uncertainty for simulation-based forecasts

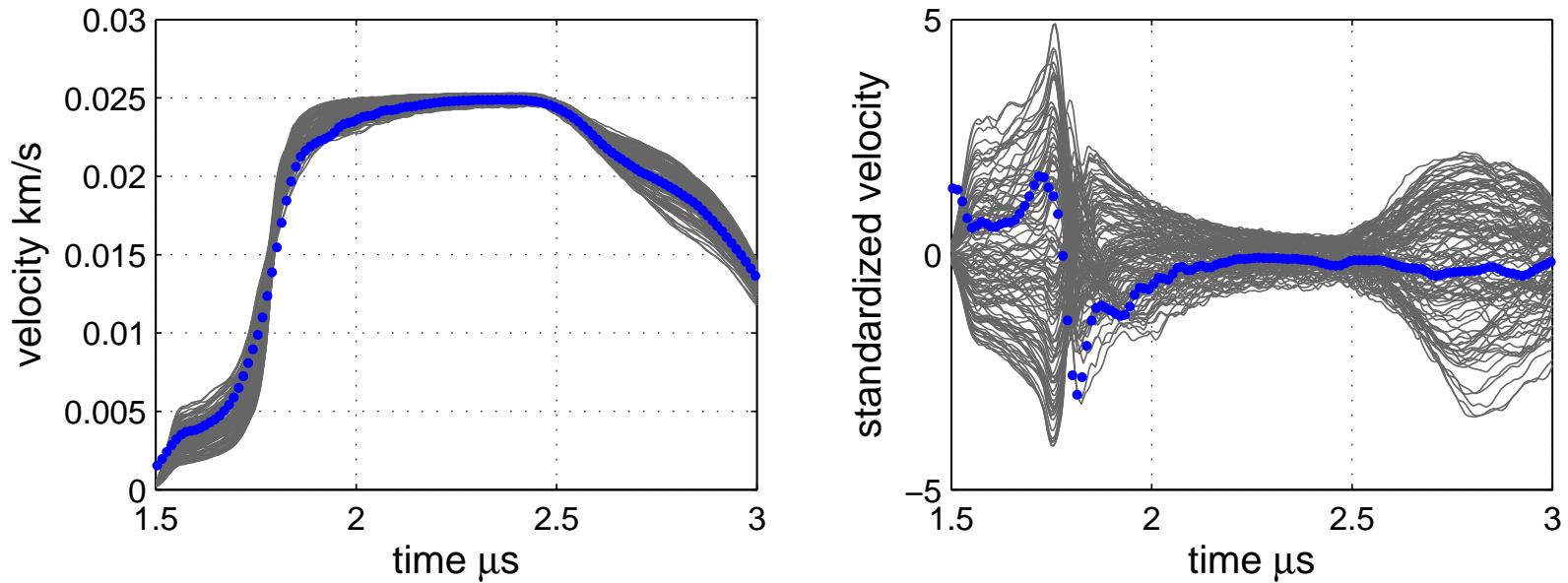
- Simulation-based predictions accumulate uncertainty due to:
 - parameter/calibration uncertainty;
 - simulator discrepancy/inadequacy;
 - observation error in data;
 - sparseness in data
- Limits on dimensionality? Have dealt with up to 20-dimensional θ .
- Statistics typically uses the wrong model (eg. a regression line) to explain data. So the framework is nothing new, in principle.
- Calibrating a model with substantial inadequacy? only slight inadequacy?
- The slowness of the simulator and high dimensionality make things difficult.
- Extrapolation is often a goal in such investigations. Generally, the closer to reality the simulator is, the more it can be trusted for extrapolation. Can this be more rigorously formalized?

Application: Ta Flyerplate Experiment



- PTW model governs features on the visar velocity profile.
- Use principal components (EOF's) to deal with high dimensionality.

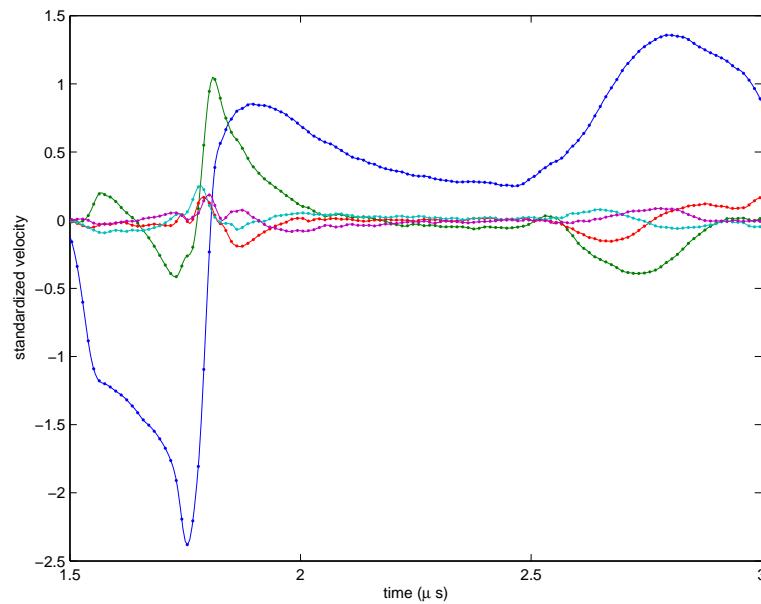
Simulations and Data



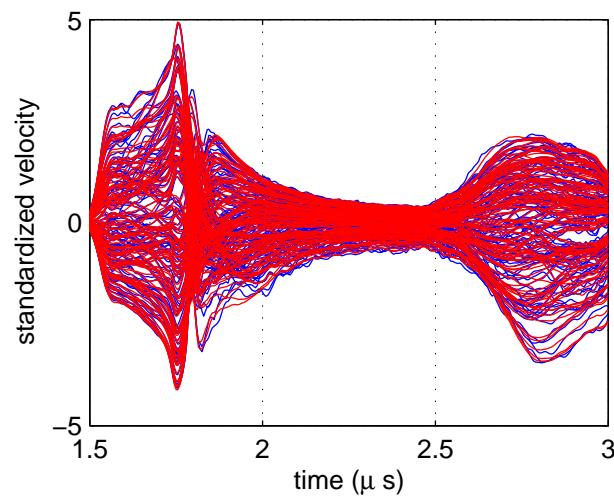
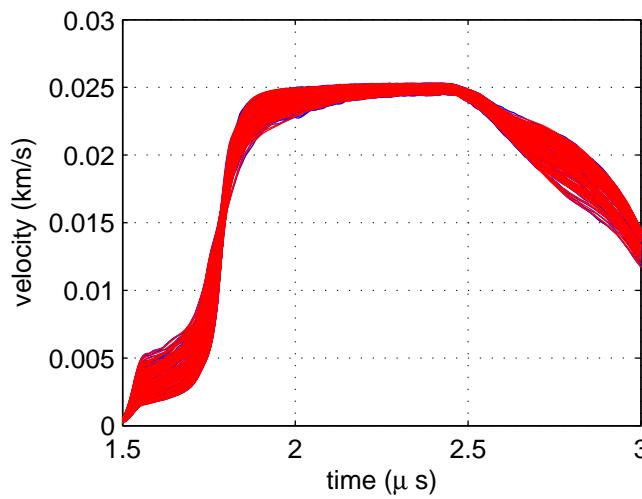
PTW calibration parameters with input domains

Parameter	Description	Domain	
		Min	Max
θ_0	Initial strain hardening rate	0.00627	0.00973
κ	Material constant in thermal activation energy term — relates to the temperature dependence	0.39316	0.96899
$-\log(\gamma)$	Material constant in thermal activation energy term — relates to the strain rate dependence	7.691	15.355
y_0	Maximum yield stress (at 0 K)	0.00689	0.01147
y_∞	Minimum yield stress (\sim melting)	0.00112	0.00182
s_0	Maximum saturation stress (at 0 K)	0.00455	0.03061
s_∞	Minimum saturation stress (\sim melting)	0.00281	0.00435

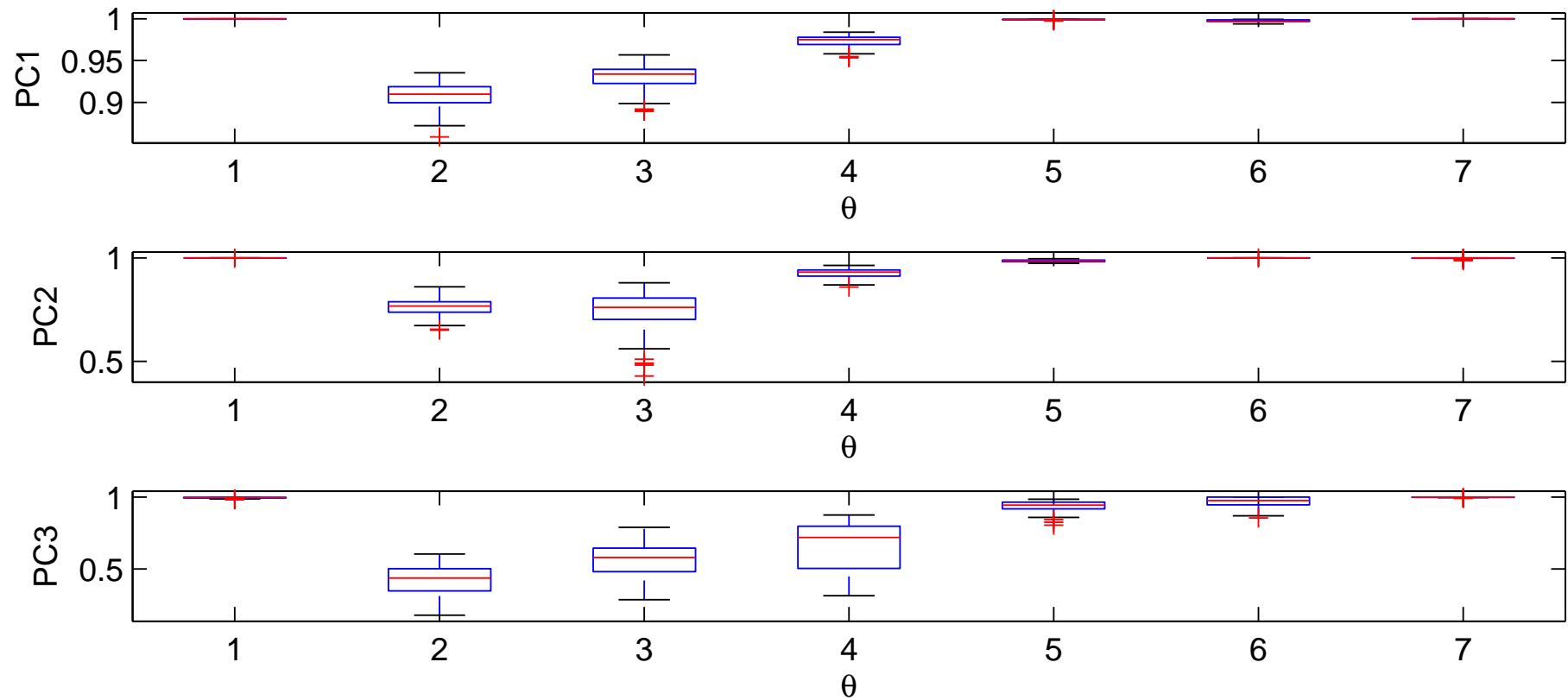
PC representation of simulation output



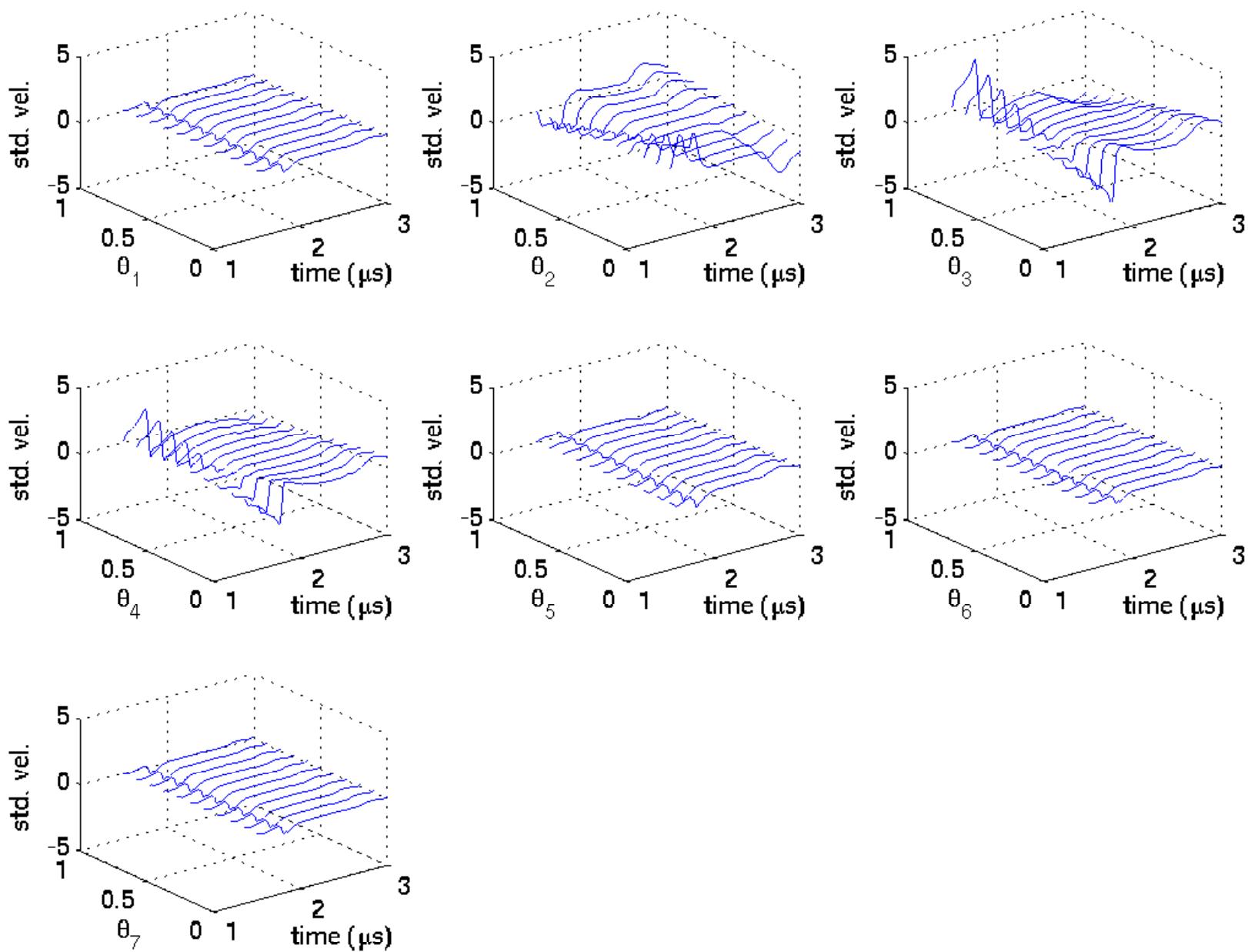
fit using $p_\eta = 5 - 75\%$ of variation



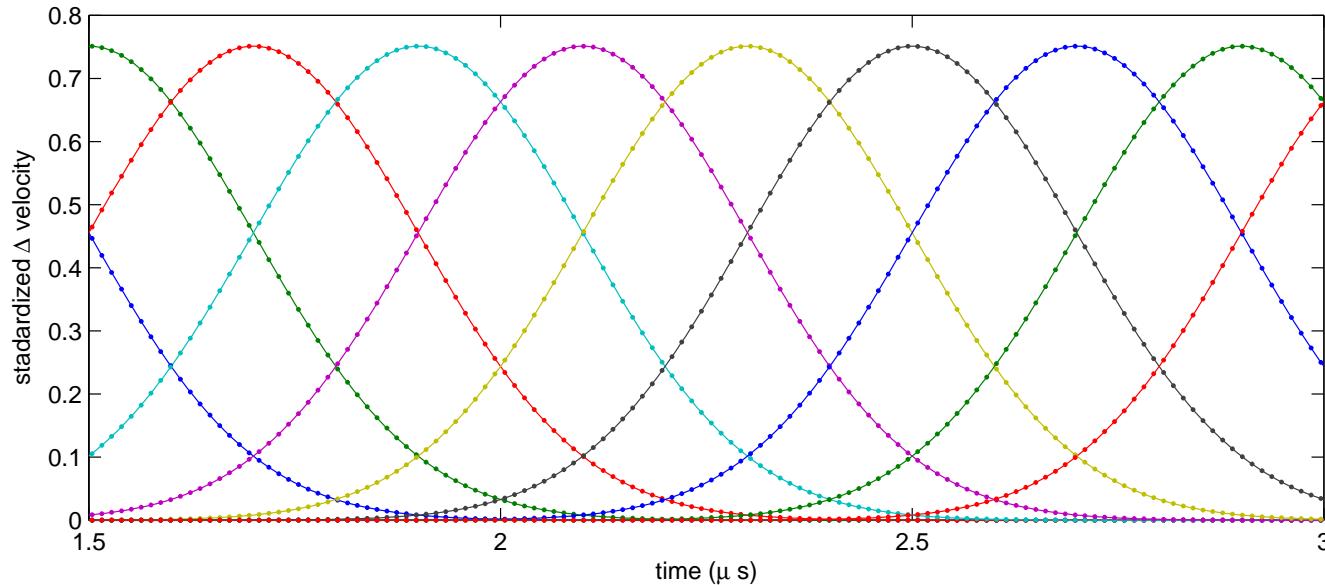
Marginal Posteriors for spatial correlation parameters ρ_{wij}



PC - based sensitivities



Discrepancy basis



Represent discrepancy $\delta(x)$ using basis functions and weights

Here $d_j(s)$ is normal density cetered at spatial location ω_j :

$$d_j(s) = \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2}(s - \omega_j)^2\right\}$$

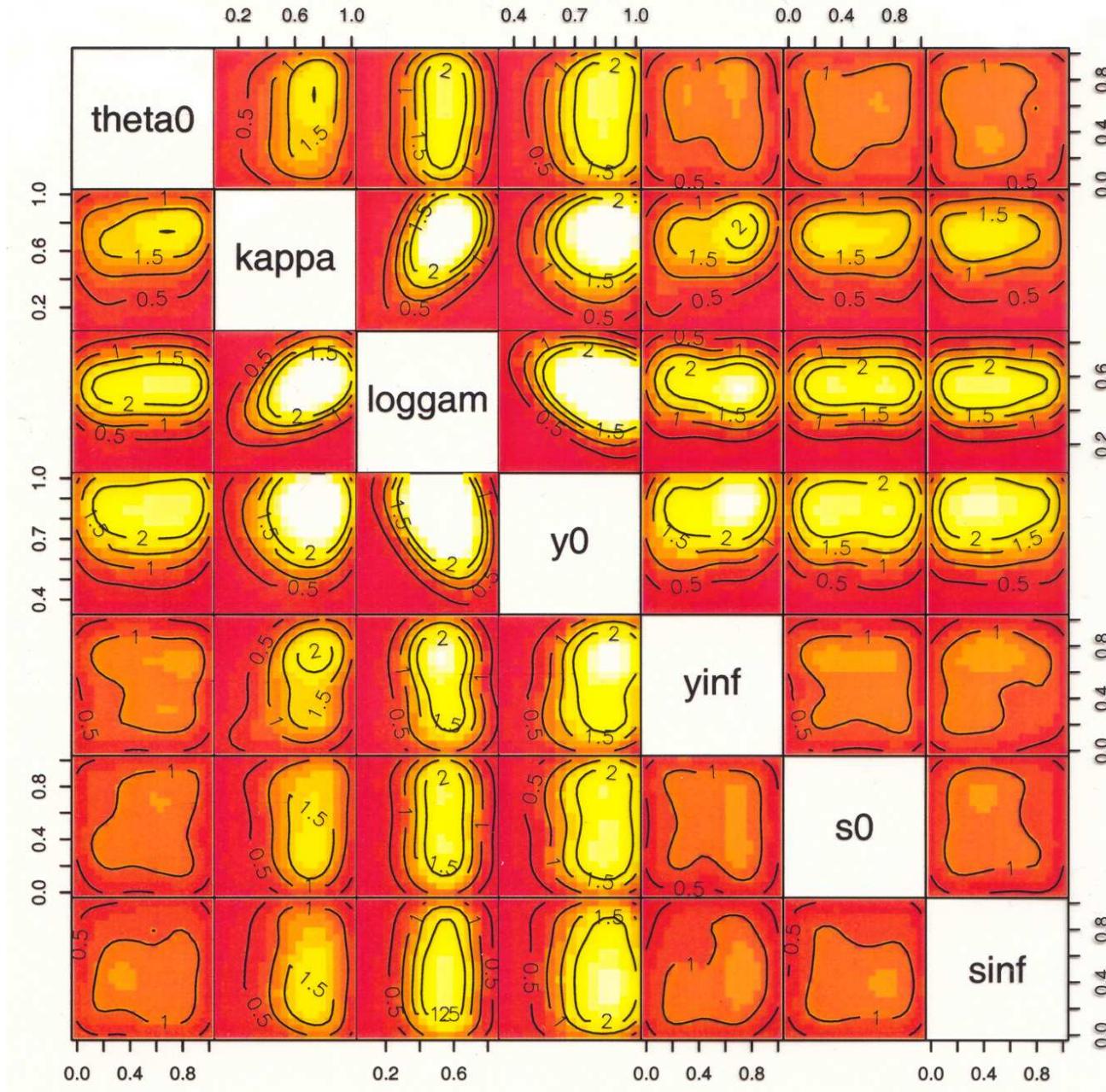
$$\text{set } \delta(s) = \sum_{j=1}^{p_\delta} d_j(s) v_j \text{ where } v \sim N(0, \lambda_v^{-1} I_{p_\delta}).$$

Can represent $\delta = (\delta(s_1), \dots, \delta(s_n))^T$ as $\delta = Dv$ where

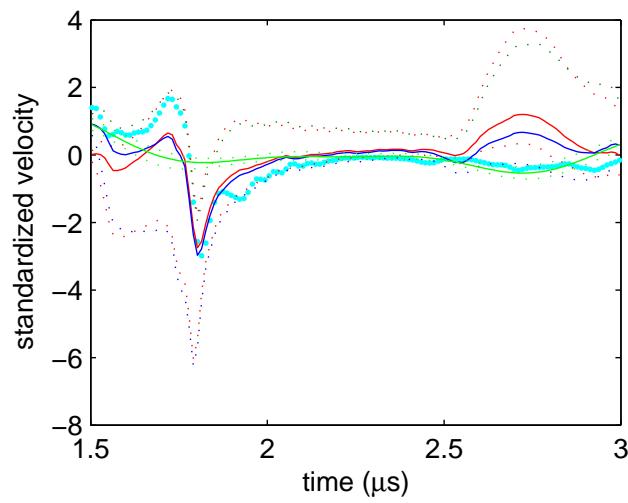
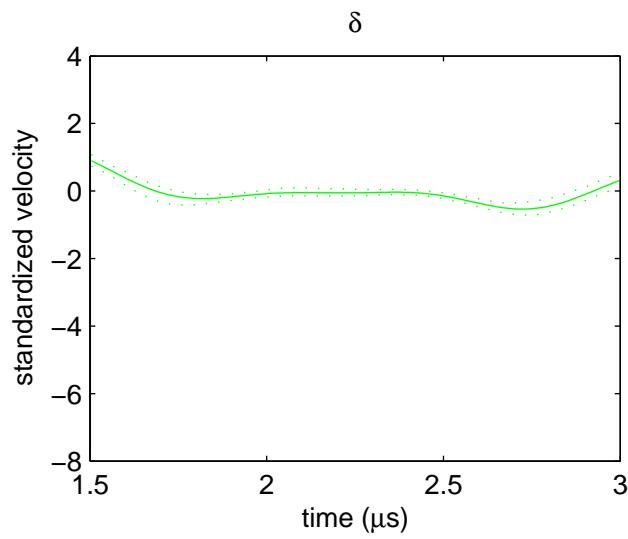
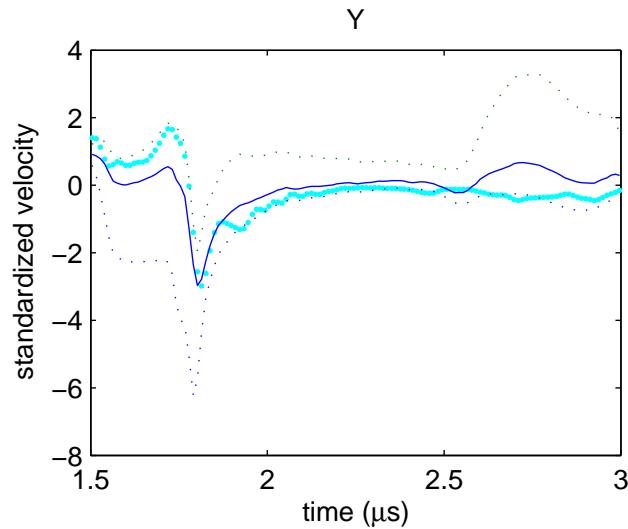
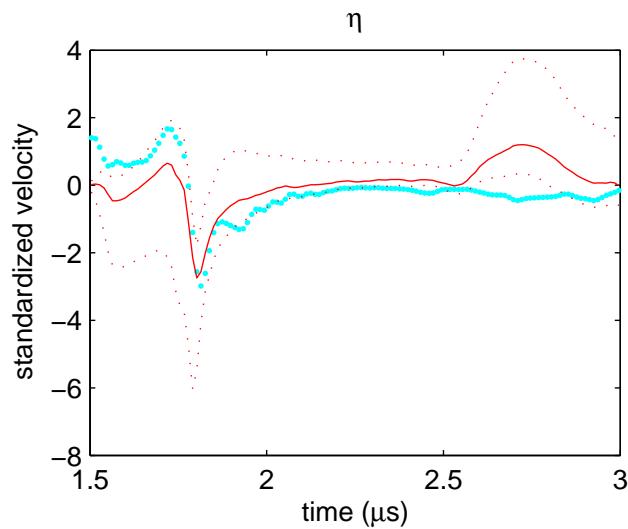
$$D_{ij} = d_j(s_i)$$

$p_\delta = 10$ basis functions over t .

posterior distribution for PTW parameters



posterior predictive distribution for trace



posterior predictive distribution for trace (original scale)

